TRAFFIC FLOW MODELING IN HIGHWAY NETWORKS

by

Tungsheng Yu

Project submitted to the faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

Computer Science

©Tungsheng Yu and VPI & SU 1992

APPROVED:

Dr. Clifford A. Shaffer, Chairman

Dr. Lee D. Han
Dr. Osman Balci

September, 1992
Blackburg, Virginia
TRAFFIC FLOW MODELING IN HIGHWAY NETWORKS

by

Tungsheng Yu

Committee Chairman: Dr. Clifford A. Shaffer
Department of Computer Science

(ABSTRACT)

The emergence of the Advanced Traffic Management System poses new challenges in traffic flow modeling of urban areas. The motivation of this project is to produce working freeway traffic simulations within a reasonable time-scale.

This project describes a hybrid traffic modeling approach, which is a combination of microscopic and macroscopic traffic modeling techniques. The traffic stream is composed of individual vehicles, while the interactions in the traffic stream are modeled macroscopically using the average speed-density relationship.

All existing freeway simulation models use the time-driven approach which advances the simulation clock after each fixed time-slice. However, this approach has a limitation of capturing the dynamic nature of traffic flow. The project proposes a new assumption of vehicular movement. This assumption leads to an easy implementation of a traffic simulation model using the event-driven approach which advances the simulation clock from one event to the next. The event-driven traffic model provides a new tool for the development of dynamic freeway simulation/assignment models.

A number of experimental results are provided from an empirical comparison of the event-driven approach versus the time-driven approach. These results indicate that the event-driven simulation model is competitive with the time-driven simulation model in both accuracy and efficiency. Finally, specific potential directions for future research are pinpointed.
ACKNOWLEDGEMENTS

I wish to extend my sincere thanks to Dr. Clifford A. Shaffer, my advisor, for his immense help and excellent guidance throughout this work. This project could not have been completed without him. I am very grateful for Dr. Antoine G. Hobeika’s financial support. I am also indebted to my committee (Drs. Shaffer, Balci, and Han) for reading this project and offering their helpful insight.

I would also like to take this opportunity to thank Dr. R. Sivanandan and my fellow graduate students Bae, Kaan, Kim, and Sam at the University Center for Transportation Research, Virginia Tech, for their advice and consultation. Thanks to my friends Dr. Joseph Derrick, Jay Wang, Sandeep Somaiya, Yaoen Zhang, Mark Lattanzi, and Colin Klipsch, for their experience and lasting friendship. All of their contributions had a positive influence upon this research.

Finally, I must thank my parents for the support and encouragement they have given me throughout my education. A very special thanks to my wife, Heng Shi, who has displayed tremendous patience and sacrifice while her husband completed this work.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>ii</td>
</tr>
<tr>
<td>ACKNOWLEDGEMENTS</td>
<td>iii</td>
</tr>
<tr>
<td>TABLE OF CONTENTS</td>
<td>iv</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>vii</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>viii</td>
</tr>
<tr>
<td>1 INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Problem Statement</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Simulation in Traffic Studies</td>
<td>2</td>
</tr>
<tr>
<td>1.3 Background</td>
<td>5</td>
</tr>
<tr>
<td>1.3.1 Terminology</td>
<td>5</td>
</tr>
<tr>
<td>1.3.2 System Identification and Model Scope</td>
<td>6</td>
</tr>
<tr>
<td>1.3.3 Queuing Systems</td>
<td>7</td>
</tr>
<tr>
<td>1.4 Project Organization</td>
<td>7</td>
</tr>
<tr>
<td>1.5 Summary of Contributions</td>
<td>8</td>
</tr>
<tr>
<td>2 BUILDING A TRAFFIC SIMULATION</td>
<td>9</td>
</tr>
<tr>
<td>2.1 Time-Driven Approach</td>
<td>10</td>
</tr>
<tr>
<td>2.2 Event-Driven Approach</td>
<td>10</td>
</tr>
<tr>
<td>2.3 Simulation Models as a Black Box</td>
<td>11</td>
</tr>
<tr>
<td>2.4 Validation and Verification</td>
<td>13</td>
</tr>
</tbody>
</table>
CONTENTS

2.5 Transient Removal and Stopping Criteria .......................... 14

3 MODELING NETWORK TRAFFIC FLOW .............................. 16
  3.1 Previous Work in Traffic Modeling ............................... 17
    3.1.1 Microscopic Models of Traffic Flow ......................... 17
    3.1.2 Macroscopic Models of Traffic Flow ....................... 18
  3.2 A Hybrid Modeling Approach .................................... 20
    3.2.1 Vehicle Generation ...................................... 21
    3.2.2 A Flexible Network Representation ....................... 22
    3.2.3 Vehicular Movement: Two Assumptions .................... 24
  3.3 Summary .................................................... 26

4 IMPLEMENTATION OF TWO TRAFFIC MODELS ....................... 28
  4.1 A Modified Time-Driven Simulation ............................ 28
  4.2 An Event-Driven Simulation .................................. 32
  4.3 Computational Complexities of EDS and MTDS .................. 34
  4.4 Summary .................................................... 36

5 EXPERIMENTAL COMPARISONS BETWEEN EDS AND MTDS ............. 37
  5.1 Simulation Experimentation .................................... 38
    5.1.1 Scenario 1: A Single Highway ............................ 38
    5.1.2 Warmup Periods ....................................... 39
    5.1.3 Simulation Results .................................... 43
  5.2 Scenario 2: A Merging Highway ................................ 47
  5.3 Scenario 3: A Sample Network ................................ 49
  5.4 Conclusions ................................................ 50

6 CONCLUSIONS AND FUTURE WORK .................................. 54
  6.1 Areas of Future Research ..................................... 55
    6.1.1 A Concurrent Simulation Algorithm ....................... 55
CONTENTS

6.1.2 An Optimistic Distributed Simulation Approach ............... 57
6.1.3 A Visual Traffic Simulation Support Environment ............. 57
6.1.4 A Real-Time Traffic Simulation .......................... 58

REFERENCES .......................... 60

VITA .................................. 65
LIST OF FIGURES

2.1 Black-Box View of a Simulation Model ........................................... 12
3.1 A Division of Link \( L_i \) ................................................................. 23
4.1 A Flowchart of MTDS ................................................................. 30
4.2 Vehicle Moving Process ............................................................... 31
4.3 A Section of Freeway ................................................................. 31
4.4 A Flowchart of EDS ................................................................. 35
5.1 Scenario 1 — a Single Highway ................................................... 38
5.2 Comparison of Averaged Travel Time across Replications .............. 40
5.3 Comparison of Warmup Period for Total Travel Time .................. 41
5.4 Effect of Arrival Rate on Warmup Period .................................... 42
5.5 Scenario 2 — a Freeway with two Ramp Junctions ....................... 47
5.6 Scenario 3 — a Sample Network ................................................. 49
LIST OF TABLES

3.1 The Representation of a Segment ........................................... 23
4.1 The Data Structure of a Vehicle .............................................. 29
4.2 Physical Link Data ............................................................. 31
4.3 Comparison of Mean Travel Times (min) ............................... 32
4.4 The Data Structure of an Event ............................................. 33
5.1 Physical Link Data in Scenario 1 ........................................... 39
5.2 Comparison of Mean Travel Times (min) of Scenario 1 ............ 44
5.3 Comparison of Mean Densities (veh/lane/mile) of Scenario 1 ..... 45
5.4 Comparison of Effect of Arrival Rate on Major Performances of Scenario 1 . 46
5.5 Comparison of Execution Times of Scenario 1 ....................... 46
5.6 Physical Link Data in Scenario 2 ........................................... 47
5.7 Comparison of Major Performances of Scenario 2 .................. 48
5.8 Comparison of Execution Times of Scenario 2 ....................... 49
5.9 Physical Link Data in Scenario 3 ........................................... 49
5.10 Eight Routes in Scenario 3 .................................................. 50
5.11 Comparison of Major Performances of Scenario 3 ............... 51
5.12 Comparison of Traffic Volume of Scenario 3 ....................... 52
5.13 Comparison of Execution Times of Scenario 3 ....................... 52
Chapter 1

INTRODUCTION

1.1 Problem Statement

Growing traffic congestion in urban areas is drawing increasing attention from the public, government officials, and transportation professionals. Congestion results in closure of roads and reduction of traffic carrying capacity. Congestion on U.S. urban freeways, which carry nearly 30% of all traffic in urban areas, is estimated to cause 1.2 billion vehicle-hours of delay, 1.3 billion gallons of wasted fuel, and 9 billion dollars in excess user costs per year. Since urban freeway travel increases at a rate of 1.9% per year, the problem will continue to increase in severity [Lindley 1987].

Increasing the capacity of the highway by increasing the number of lanes is not always an acceptable solution. An alternative approach consists of exerting some kind of control over the flow of vehicles by means of signals, with the objective of avoiding unnecessary congestion and reducing the incident rate. Smulders [1990] addressed such freeway traffic control problems by means of the variable speed signs for the Dutch Motorway Control and Signaling System. This system allows the display of advisory speed signals to drivers and aims to improve traffic flow and safety, based on speed measurements and vehicle counts available at several locations along the freeway. Another approach is to reroute traffic through a highway network [Cremer and Fleischmann 1987 and Mahmassani et al. 1991].
This approach seems to be less well developed. The challenge is lack of a simulation model that is able to efficiently predict travel times and traffic densities for each link and each route of a network. It is the motivation of this project to solve this problem by appropriately modeling freeway flow on a computer.

There are three typical sources of congestion: (1) the general level of traffic along a link, including interactions between the traffic and the abutting land uses; (2) the junction delays and capacity restrictions due to the interactions and disruptions occurring when two or more traffic streams intersect; and (3) the adverse effects of incidents such as traffic accidents, road work, and sporting events. However, with the increasing traffic demand on a limited transportation infrastructure and the increasing sophistication of traffic control strategies, the comprehensive analysis of congestion is becoming increasingly complex. The process of describing such complex systems using only analytical or mathematical models can be difficult or even impossible. To progress beyond this point it is necessary to turn to a more sophisticated tool such as computer simulation.

In addition, it is an expensive operation to empirically study the effects of traffic management techniques on traffic flow and travel time along the highway network. Computer simulation models of traffic flow are also useful tools in this respect. Using a computer to simulate the operations of a real traffic network requires that a set of assumptions taking the form of logical or mathematical relationships be developed and shaped into a model. We can then use this model to gain an understanding of the dynamics at work in the transport network. The simulation program is evaluated numerically over a period of time, and statistical data is gathered to estimate the true characteristics of the actual system.

### 1.2 Simulation in Traffic Studies

Since the advent of electronic computers, traffic simulation was thought to be particularly useful for traffic studies. Traffic simulation is an attempt to replicate, on a computer, the sequence of events that make up a real transport system. There are many situations where
simulation provides a powerful approach for transport professionals to assess the effects of changes in the traffic system and their consequences on its efficiency. A wide variety of specific or general purpose traffic simulation models have been built. These models can be classified into four groups [Young et al. 1989] according to their area of application:

1. **Transport and traffic demand models** which provide information on the overall demand for transport infrastructure in an urban context, such as EMME/2 [Florian et al. 1979], TRANSTEP [Nairn 1984], MICROTRIPS [Johnston 1984], MINUTP [UMTA 1985], and TRANPLAN [Benham, 1986].

2. **Dense network models** which are developed for the design of traffic networks, such as SATURN [Bolland et al. 1979], MULATM [Taylor 1988a,b], NETSIM [Lieberman et al. 1977 and Andrew et al. 1988], TRAFFICQ [Logie 1979], CONTRAM [Leonard 1978 and Leonard et al. 1989], TRANSYT [Viagent et al. 1980 and Wallace 1983], and METANET [Morin et al. 1991].

3. **Traffic route design or guidance models** which are developed for studying traffic movements on links, such as FREFLO [Payne 1979], INTRAS [Wicks et al. 1980], PASSER II [Rogness and Messer 1983], FREQ [Imada and May 1984], KONOS [Michalopoulos 1984], AUTOGUIDE [Jeffery 1987], and ROGUS [Harris et al. 1992].

4. **Intersection design models** which deal with the design of intersections, such as TEXAS [Lee et al. 1978], SIDRA-2 [Akcelik 1985], PICADY 2 [Semmens 1985], and INSECT [Nairn and Partners 1986].

There are a number of reasons for using simulation in traffic engineering in preference to other analysis techniques. Some prime reasons are as follows:

1. Reduce the risk and uncertainty associated with a given traffic system.

2. Allow experimentation to be made without disruptions to the existing system.

3. Allow concepts to be tested prior to installation.

4. Run the simulated system at speeds much greater than would be attainable in the real world.

5. Model any complex traffic system at any level of desired detail.

6. Model more realistically, with less need for dubious assumptions required by analytical models to make a complex system tractable.

7. Allow both input and output to be more extensive and more transparent.
8. Give insight to the true operating characteristics of the system being modeled.

However, the practice of simulation requires knowledge of a range of topics, including a detailed understanding of the situation to be modeled, insights into the modeling process, skills in correctly representing a model in computer-processable terms, data collection and analysis, design and conduct of experiments, analysis of output, and interpretation of results in the context of the situation that was modeled. Thus, there are a few disadvantages inherent with the use of simulation such as:

1. Time Consuming: Data collection, model development, analysis, and report generation will require a considerable amount of time.

2. Yields only approximate answers: Since a model is an approximation of the system being studied, it is computationally intractable to develop a full-scale representation for network transportation. In addition, simulation usually relies on the use of random number generators to provide model input, there is some uncertainty associated with the output that must be dealt with statistically. That is, all outputs are only estimates of the true behavior of the system.

3. Difficult to validate: When measurement of the real system does not exist, validation can become a formidable task.

The use of simulation can reduce the risk involved with implementing a new transport system or changing an existing one. However, unless simulation is used prudently, the possibility of developing an invalid model exists. There are several common pitfalls along the way to successful completion of a traffic simulation study [Law and Kelton 1991]:

1. Failure to have a well-defined set of objectives at the beginning of the simulation study.

2. Use of the false assumptions in model construction.

3. Selection of inappropriate level of model detail.

4. Failure to have people with operations-research and statistical training on the modeling team.

5. Failure to account correctly for sources of randomness in the actual system.

6. Making a single replication of a particular system design and treating the output statistics as the “true answers”.

4
7. Poor selection of simulation software. Some commercial simulation software may not be well documented or implement the modeling logic desired.

In order to insure the success of a simulation project, care must be taken to avoid these pitfalls. To aid in the decision regarding credibility, McHaney [1991] has pinpointed six symptoms of a sick simulation. That is, if any of these symptoms is present, the credibility of the model should be questioned. The symptoms are as follows:

1. Inaccurate input data.
2. Improper run times.
3. Inconsistent output.
4. Vague references to assumptions.
5. Incomplete output.
6. No animation or detailed trace of events.

1.3 Background

1.3.1 Terminology

An important part of any discipline is developing a grasp of the terminology commonly used by its practitioners. Listed below are a few words and phrases with special meaning that are used in this project.

- **Model and Simulation.** A *model* is an abstraction of reality. *Modeling* is the process of building a model. *Simulation* is the process of defining, creating, and studying a model with the intention of drawing inferences and forming conclusions as to the behavior of the actual system. *Computer simulation* not only includes building a mathematical model for a system but also conducting experiments with this model on a computer.

- **State Variables and Events.** The variables whose values define the state of the system are called *state variables*. An *event* denotes a change in the state of the system being simulated, and contains a time-stamp that indicates when this change occurs. There are two distinct types of events:
1. a **bound** (*i.e.*, unconditional) **event** is one whose occurrence is predictable and can thus be scheduled;

2. a **conditional event** is one whose occurrence is dependent upon the fulfilment of certain conditions.

- **Deterministic and Probabilistic Models.** If the output of a model can be predicted with certainty, it is a **deterministic model.** A **probabilistic model,** on the other hand, gives a different result on repetitions for the same set of input parameters.

- **Static and Dynamic.** A model in which time is not a variable is called **static.** If the system state changes with time, the model is **dynamic.**

- **Discrete Simulations and Steady-State Simulations.** A **discrete simulation** is a simulation where the state of the system changes only at some discrete points in time. A variable is in a **steady state** if its average value remains the same over the time period under consideration. A **steady-state simulation** is one where each output variable will attain a steady state after a long enough period of time.

- **Simulation Clocks and Simulation Strategies.** Each simulation has a global variable called **simulation clock** representing the current value of simulated time. The simulation clock has two features different from the physical system **real time:** (1) starting with zero; and (2) stopping at some (pre-specified) time. A **simulation strategy,** also referred to as a world view or conceptual framework, is an underlying structure and organization of ideas which constitute the outline and basic frame that guide a modeler in representing a system in the form of a model [Derrick 1992].

### 1.3.2 System Identification and Model Scope

**System identification** establishes relationships between an analyzer and some part of reality. A **system** is defined as a collection of objects, their relationships and behavior relevant to a set of purposes, characterizing some relevant part of reality. That is, modeling should be a purpose-driven activity, and a model’s validity always depends on objectives and context of its application [Kreutzer 1986]. The objectives specify what questions the simulation needs to answer.

**Model scope** is defined as that portion of the system that will be represented by the model. The necessary level of detail is a determination of how closely the model has to emulate the real world system to still provide the required information [McHaney 1991]. Since a model is an approximation of the system being studied, it is not always best to develop a
full-scale representation. Modeling in too much details leads to excessive development time. On the other hand, too little detail may result in a model that does not realistically predict actual world behavior. Therefore, it is very important to the success of a traffic simulation project for us to insure that the model adequately reflects transport network operation.

1.3.3 Queueing Systems

Queues are a common aspect of any irregular process operating in an environment with limited capacity. A queueing system consists of a service facility that provides service of some kind to arriving customers. The service facility requires a certain time to serve each customer and is capable of serving only a limited number of customers at a time. If customers arrive faster than the facility can serve them, they must wait in a queue for facility. In the highway network, customers are vehicles and servers are the transportation infrastructure such as links and intersections. Congestion will occur in the system if there is sufficient irregularity. To predict and prevent congestion the system must be specified in sufficient detail, usually including the following three basic components:

- **Arrival pattern** describes how customers arrive to the system, including both the average rate of arrival for customers and the statistical pattern of the arrivals.

- **Service mechanism** specifies the condition of servers, such as whether each server has its own queue or there is one queue feeding some servers, how many customers can be served at one time, and how long each service takes.

- **Queue discipline** refers to the rule that a server uses to choose the next customer from the queue (if any) when the server is available.

1.4 Project Organization

This chapter introduces background material and some basic simulation concepts used in this project. Chapter 2 describes two typical discrete-event traffic simulations, and covers some simulation methodologies such as how to test a traffic simulation model and how to obtain desired results from a simulation.
Chapter 3, 4 and 5 form the core of the project. Chapter 3 reviews the previous work in modeling network traffic and then focuses on describing a hybrid traffic modeling approach. Chapter 4 describes the implementation of the time-driven and event-driven traffic simulation models. Chapter 5 focuses on comparing these two models by some typical experiments.

Finally, Chapter 6 presents conclusions on this research and pinpoint some potential directions for future research.

1.5 Summary of Contributions

This project gives the following eight significant contributions:

1. Summarizes previous work in traffic simulation, including traffic model classification, traffic simulation advantages and disadvantages, and pitfalls in traffic modeling.

2. Builds the fundamental relationship among volume, density and speed from Little's Law.

3. Presents a hybrid traffic modeling approach, which is a combination of microscopic and macroscopic traffic simulation techniques.

4. Modifies and implements a time-driven simulation model of network traffic flow.

5. Designs, creates and implements an event-driven simulation model of network traffic flow, which opens a new simulation environment for network traffic study.

6. Develops a series of traffic simulation routines that range from vehicle generation to output analysis. These routines are flexible and well-documented.

7. Presents three typical scenarios for traffic simulations that can be combined to form more complicated traffic networks, and conducts experimental comparisons between the event-driven and time-driven simulations.

8. Proposes a concurrent algorithm for parallel traffic simulation and some suggestions for future work.
Chapter 2

BUILDING A TRAFFIC SIMULATION

Many traffic simulations use *discrete-event simulation models*. Although these traffic simulations have their different objectives and utilities, their models still share a number of common components and a logic organization for these components. These components include:

- Data Structures for System State Variables: These are global variables necessary to describe the state of system.
- A Simulation Clock: To keep track of simulation process dynamically.
- Event Routines: Each event is simulated by its routine. These routines update the system state variables and schedule other events.
- Initialization Routines: These set the initial state of the system state variables and initialize various random-number generation streams.
- Input Routines: These get the model parameters, such as physical features for each link segment.
- Statistical Records: These data structures store statistical information used later to compute the desired measures of performance.
- Report Generators: These are the output routines executed at the end of the simulation. They calculate the final result from the statistical records and print in a specified format.
• Main Program: This brings all the routines together. It calls input routines, initializes
the simulation, executes various iterations, and finally, calls the report generators.

In practice, the selection of a simulation strategy is critical for the modeler. The ap-
propriate selection enables a modeler to implement a simulation with significantly reduced
complexity [Balci 1988]. There are two selection guidelines for modelers:

1. Understand the problem domain, since some specific problems match one simulation
strategy better than another.

2. Consider simulation model characteristics such as maintainability, efficiency, modifi-
ability, reusability, and ease of development.

There are two typical approaches that may be used in traffic simulations: *time-driven*
and *event-driven* approaches.

2.1 Time-Driven Approach

The *time-driven* approach updates the simulation clock at discrete points in time to keep
track of the physical positions of vehicles on the roadway. The simulation begins with the
traffic system in its initial state at time $T_0$. The time clock is then advanced one increment,
$\Delta T$, to $T_0 + \Delta T$, and the updated state of the system is determined. This process is
repeated until the simulation is complete. The time-driven approach is most appropriate
when there are a large number of events to be considered or events are not discrete. We
refer to a traffic simulation as a *Time-Driven Simulation (TDS)* if the simulation uses the
time-driven approach.

2.2 Event-Driven Approach

The *event-driven* approach advances the simulation clock only when the next event occurs.
We refer to a traffic simulation as an *Event-Driven Simulation (EDS)* if the simulation uses
the event-driven approach.
Event scheduling is the most commonly used simulation strategy when implementing an EDS in a high-level programming language. Under this simulation strategy, the modeler considers the system of interest to be described in terms of events. An event routine, a set of actions following a state change in the system, is associated with an event. After initialization, the next event is determined and executed. The simulation clock is updated at this time. The event routine associated with the next event is then executed. When applicable, the conditions for termination of the simulation are checked. If the termination condition is satisfied, the simulation output is produced and the simulation ends. Otherwise, the (new) next event is selected and the algorithm repeats itself.

Event routines associated with unconditional events can be explicitly scheduled with the implementation of an event-set algorithm, an algorithm that facilitates the selection and processing of events. Two operations are performed frequently on this set — one to insert new events in the set and one to find the next event and remove it from the set. The choice of the data structure used to maintain the event set affects the processor time required for the two operations. A popular event-set algorithm is to keep an ordered linked list of future event records. Each record contains the time at which the event should occur and a pointer to the code that must be executed at that time. Conditional events occur at the satisfaction of some set of conditions that cannot be predicted in advance. The checks for conditional events are contained within unconditional event routines.

The simulation termination condition is usually based on the number of departures in a queueing network, and thus needs to be checked after executing the events associated with departures. However, other events (e.g., arrivals) do not affect this termination condition, so the termination condition does not need to be tested after completion of these routines.

2.3 Simulation Models as a Black Box

To consider a simulation as part of a decision-making process, it is helpful to regard the simulation model as a black box that frees us from the complicated internal logic of the
simulation model. Figure 2.1 shows a simulation as a black box together with the inputs to and the outputs from the model. Generally, the inputs to the model include the physical and operational features of a highway network such as:

1. Arrive pattern of each source
2. Number of links
3. Length of each link
4. Number of lanes for each link
5. Free flow speed for each link
6. Capacity for each link
7. Number of intersections
8. Control pattern for each intersection
9. Incident pattern, which describes how incident occurs and disappears
10. Geometric features of each link

The outputs from the simulation include the following three network characteristics:
1. Travel cost, including mean travel time for each link and each route.
2. Average density for each link and each freeway
3. Mean traffic volume for each link.

From the black-box point of view, we find Little's law [Little, 1961] that can be used to relate the mean number of vehicles with the mean travel time on a network (link):

\[ \text{Mean vehicle number} (N) = \text{Traffic arrival rate} (q) \times \text{Mean travel time} (t) \]

where traffic arrival rate \((q)\) is the average number of vehicles passing a source in a link (segment) per unit of time, or traffic volume at that source.
Let $L$ be the length of a link (segment), then the above formula is equivalent to the following equation:

$$q = \frac{N}{L} \times \frac{L}{t}$$  \hspace{1cm} (2.1)

Equation 2.1 actually describes the following fundamental relationship of the three basic traffic characteristics (volume, density and speed):

$$q = KV$$  \hspace{1cm} (2.2)

where $V$ is the average speed of vehicles along the link (segment) at some instant, and $K$ is the mean density, or the average number of vehicles occupying a unit length of lane.

Notice that Little's law can be used as long as the number of the vehicles entering the network (link) is equal to those exited from the network (link). That is, inside the network (link) no new vehicles are created and no vehicles are lost.

## 2.4 Validation and Verification

*Validation* is the process of making sure that the simulation model accurately represents the system being modeled [Balci 1991]. Validation deals with building the right simulation model. A valid model would possess a sufficiently accurate representation of the problem.

*Verification* is the process of making sure that the simulation model has been transformed from one form (i.e., conceptual model) into another (i.e., working simulation model) as intended with sufficient accuracy. Verification deals with building the simulation model right. Sometimes, model verification and model validation are referred to as model testing [Balci 1991].
2.5 Transient Removal and Stopping Criteria

After model development is complete, we need to decide how many of the initial observations should be discarded to ensure that the model has reached a steady state and how long to run the simulation. These two issues are called transient removal and stopping criterion, respectively [Law and Kelton 1991].

Our network traffic flow simulation is intended to be a steady-state simulation. Only the steady-state performance, that is, the performance after the system has reached a stable state, is of interest. The results of the initial part of the simulation should not be included in the final computations. This initial part is called the transient state, and the time interval associated with the transient state is called the warmup period. The main difficulty of transient removal is that it is impossible to define exactly what constitutes the transient state and when that transient state ends. All methods for transient removal are therefore heuristic.

For traffic simulations, we find that Welch's procedure [Welch 1983] is the most useful method to determine the warmup period. To reduce the effect of randomness, the method requires first averaging across several replications. Each replication consists of a complete run of the simulation with no change in input parameter values. The replications differ only in the seed values used in the random-number generators. Averaging across the replications results in an average trajectory $\bar{Y}_i$. To further smooth out $\bar{Y}_i$, the method defines the moving average $\bar{Y}_i(w)$, where $w$ called window is a positive integer less than half size of each replication.

Suppose there are $n$ replications of length $m$ each. Let $Y_{ji}$ denote the $i$th observation in the $j$th replication ($j = 1, 2, ..., n; i = 1, 2, ..., m$). Welch's procedure consists of the following three steps:

1. Get an average trajectory across replications: $\bar{Y}_i = \frac{1}{n} \sum_{j=1}^{n} Y_{ji}$, $i = 1, 2, ..., m$

2. Compute the moving average $\bar{Y}_i(w)$ as follows:
\[ \overline{Y}_i(w) = \begin{cases} \frac{1}{2i-1} \sum_{s=-i}^{i-1} \overline{Y}_i+s & \text{if } 1 \leq i \leq w \\ \frac{1}{2w+1} \sum_{s=-w}^{w} \overline{Y}_i+s & \text{if } w + 1 \leq i \leq m - w \end{cases} \]

3. Plot \( \overline{Y}_i(w) \) for \( i = 1, 2, ..., m - w \) and choose \( l \) to be that value of \( i \) beyond which \( \overline{Y}_1(w), \overline{Y}_2(w), ... \) appears to have converged. The value of \( l \) is the warmup period.

It is important that the length of the simulation be properly chosen. If the simulation is too short, the results may be highly variable. On the other hand, if the simulation is too long, computing resources and manpower may be unnecessarily wasted. Generally, the simulation should be run until the confidence interval for the mean response narrows to a desired width. If the sample mean is \( \overline{x} \) and its variance is \( Var(\overline{x}) \), a 100(1 - \( \alpha \))% confidence interval for the mean is given by \([\overline{x} - z_{1-\alpha/2} Var(\overline{x})], \overline{x} + z_{1-\alpha/2} Var(\overline{x})\] , where \( z_{1-\alpha/2} \) is the \((1 - \alpha/2)\)th quantile of a unit normal variate. Some commonly used normal quantiles are: \( z_{0.95} = 1.645, z_{0.975} = 1.960, z_{0.99} = 2.326, \) and \( z_{0.995} = 2.576 \).

We use a statistical method called replication/deletion approach to obtain an estimate and confidence interval for \( \overline{x} \). The method consists of conducting \( n \) replications of size \( m + l \) each, where \( l \) is the warmup period determined by Welch’s procedure and \( m \) is much larger than \( l \) [Law and Kelton 1991]. The remaining steps are as follows:

1. Compute a mean for each replication: \( \overline{x}_j = \frac{1}{m} \sum_{i=l+1}^{l+m} x_{ji}, j = 1, 2, ..., n \)

2. Compute an overall mean for all replications: \( \overline{\overline{x}} = \frac{1}{n} \sum_{j=1}^{n} \overline{x}_j \)

3. Calculate the variance of replicate means: \( Var(\overline{x}) = \frac{1}{n-1} \sum_{j=1}^{n} (\overline{x}_j - \overline{\overline{x}})^2 \)

The confidence interval for \( \overline{x} \) is \([\overline{x} - z_{1-\alpha/2} Var(\overline{x})], \overline{x} + z_{1-\alpha/2} Var(\overline{x})\]. Notice that this method requires discarding \( nl \) initial observations, and the confidence interval width is inversely proportional to \( \sqrt{mn} \). Thus, a narrower confidence interval can be obtained equally well by increasing either \( m \) or \( n \). However, to reduce the waste (\( nl \) initial observations), we usually keep \( n \) fairly small (e.g., 10) and take \( m \) big enough to obtain the desired confidence.
Chapter 3

MODELING NETWORK TRAFFIC FLOW

A congestion problem may be analyzed as follows:

1. Identify the congestion influence area.
2. Collect data on network geometries and origin-destination demands.
3. Assign traffic and simulate different control strategies.
4. Evaluate alternatives and select the best of traffic control strategies.

The core of this solution is the third part. Most assignment models proposed in the literature are made up of different modules simulating demand (drivers’ choice behavior) and supply (network flows and costs). A static assignment model cannot deal with variations in demand and supply. Since traffic flow is always fluctuating, it is difficult to determine precise travel time before a trip is made. However, a number of experiments have shown that traffic can be brought much closer to the equilibrium [Sheffi, 1985] if adequate information on the route choice is given to drivers. Therefore, a dynamic traffic model is required and must be fast enough to provide predictive information under changing conditions [Ben Akiva et al. 1991]. This chapter addresses the modeling methodology involved in producing working freeway simulations within an efficient computing environment.
3.1 Previous Work in Traffic Modeling

Dynamic modeling of network traffic is generally considered to be a complex problem. The challenge is lack of a perfect modeling tool that would reflect a comprehensive theory covering (in a consistent fashion) all aspects of vehicular movement in the network. Such a theory remains a remote prospect, even to the optimists among traffic theorists [Young et al. 1989]. Theoretical studies in the past introduced a number of simplifications such as:

1. Steady-state instead of time varying conditions [Payre and Thompson 1974].
3. Fixed route choice or fixed traffic demand at the origins [Gazis 1974, Bolland et al. 1979 and Papageorgiou 1990].

Traffic simulation models can be typically divided into two categories in terms of their representation of traffic: (1) microscopic models that consider traffic as individual vehicles, and simulate their trajectories as they traverse the roadway; and (2) macroscopic models that treat traffic as vehicle flows (i.e., platoons), and simulate the flow behavior as a whole.

3.1.1 Microscopic Models of Traffic Flow

Microscopic models of traffic flow are derived from the behavior of the individual driver and seek to describe his dynamic reaction to various external influences. There are two dimensional reactions for the driver: (1) to accelerate or brake in an effort to maintain his desired speed without running into the vehicle ahead; and (2) to change lanes.

Work on the first aspect has largely concentrated on car-following models, which have the general form [Young et al. 1989]:

\[ a_n(t + \tau) = \frac{l_n(V_{n-1}(t) - V_n(t))|V_{n-1}(t) - V_n(t)|^{m-1}}{|P_{n-1}(t) - P_n(t) - S_{n-1}|^p} \]

where \( a_n(t + \tau) \) is the acceleration of vehicle \( n \) at time \( t + \tau \); \( \tau \) is reaction time; \( l_n, m, \) and \( p \) are parameters that need to be estimated; \( S_n \) is the effective length of vehicle \( n \); and, \( P_n(t) \) and \( V_n(t) \) are the location and speed of vehicle \( n \) at time \( t \), respectively.
Although car-following models have served well in many situations, they are not well-suited for use in traffic flow simulation. They are essentially designed as continuous functions, and to maintain stability in a simulation environment where position, speed and acceleration are updated at discrete points in time, it is desirable for the interval between successive recalculations to be a fraction of the reaction time. This requires storing large quantities of historical data and takes considerable computing time to evaluate general power functions.

However, lane-changing models are much more complex than car-following models because the decision to change lanes depends on a number of objectives, and at times they may conflict. There are lack of well-defined lane-changing models.

The most widely-used microscopic model is NETSIM which is a stochastic simulation model of traffic movement on networks [Lieberman et al. 1977 and Lieberman 1981]. The network is described as a set of unidirectional links and nodes. The simulation logic uses the time-driven approach with a one second time-slice. Vehicle motion is governed by car-following, lane changing and queueing algorithms, vehicles being randomly assigned characteristics on entry to the network. A recent version of NETSIM has the feature of graphic input/output displays [Andrew et al. 1988]. However, NETSIM still has “bugs” and needs to be refined.

### 3.1.2 Macroscopic Models of Traffic Flow

Macroscopic models typically consider traffic as composed of the movements of platoons. A platoon is a group of vehicles, traveling at short headways and moving at roughly the same speed. Platoons are created at network locations corresponding to origins of transportation demand at a given time, travel through the network and disappear at their destination. The majority of existing freeway simulation models falls into the category of macroscopic models. There are three popular macroscopic models:

- SATURN is a static equilibrium assignment program [Bolland et al. 1979], based on a time-driven simulation model of intersection delays. For signalized links travel
times are estimated using a platoon dispersion approach [Young et al. 1989]. The solution algorithm requires flow/delay relationships for a link as a function of link flow. SATURN assumes the total traffic volume over time to be static.

- CONTRAM is a *time-driven* traffic model, which uses an iterative procedure to assign vehicles to their minimum cost routes through a network under a user-imposed traffic demand [Leonard 1978 and Leonard et al. 1989]. CONTRAM is a time-dependent model that predicts flows, queues and routes of vehicles at each time interval over the simulation period. CONTRAM is not based on a mathematical solution for an equilibrium equation, and so is not guaranteed to converge to a solution.

- *Macroparticle Simulation Model* (MPSM) is a deterministic, time-driven traffic model to simulate vehicular movement on freeways and arterials given time-dependent input functions [Chang et al. 1985 and Mahmassani et al. 1988].

In dynamic modeling of network traffic, MPSM has played an important role and initiated this research work. We follow to discuss MPSM in more detail.

Recently, MPSM uses a time-slice of 0.1 minute [Mahmassani et al. 1991] to update the position of vehicles. The MPSM logic, adapted from plasma physics [Leboeuf et al. 1979], is retained for traffic movement on individual links. Traffic is not treated as a compressible fluid; vehicles are moved in *macroparticles* at the prevailing local speeds, consistent with a relation between the average speed and the prevailing density. Here macroparticle means that a group of vehicles always move together like a packet. Formally, MPSM assumes that:

- Traffic demand and system capacity is fixed over the simulation period.
- Vehicles move along each link as a collection of macroparticles with fixed sizes.
- Time is discretized into small fixed time-slices, and each link can be divided into a set of segments such that all macroparticles on a segment travel at the same mean speed during each time-slice.
- Each macroparticle movement satisfies the following modified Greenshields model:

$$ V = V_0 + (V_f - V_0)(1 - K/K_j)^\alpha $$

where $V$ and $K$ are the average speed and density, respectively, in the given segment; $V_f$ and $V_0$ are the free mean speed and minimum speed, respectively; $K_j$ is the jam density; and $\alpha$ is a speed-density parameter.
These assumptions provide a macroscopic and discrete representation of traffic flow. Since MPSM assumes that the state of the network is changed at the end of each time-slice, MPSM is well-suited to implementation by the time-driven approach. However, MPSM assumptions are not realistic in that the arrivals of newly entered vehicles can affect the behavior of vehicles further along in the same segment, because the speed of all vehicles on the segment slows down when the number of newly entered vehicles is more than the number of vehicles exiting that segment during the same time-slice. The validity of the model is lowered in that the size of each macroparticle assumes as constant throughout the network. The macroparticle also makes a conflict with a dynamic assignment requirement that simulation models be applied at the individual driver level. An alternative is to formally move vehicle one by one, this has been mentioned in their recent version [Mahmassani et al. 1991]. In addition, MPSM has the following disadvantages:

- MPSM uses uniform arrival patterns and introduces new vehicles only at the end of some time-slices.
- Accuracy of simulation outputs is sensitive to the selection of a time-slice ($\Delta T$). Smaller $\Delta T$ may or may not increase accuracy but must cost much longer execution time.
- The time-slice length specifies the dynamic resolution and limitation of the model, because MPSM assumes the modeled quantities (e.g., density and congestion) remain constant with a given time-slice.

### 3.2 A Hybrid Modeling Approach

Since there is no precise traffic flow theory that can be used to build a "perfect" model, we should abandon models that prove to be too intractable, either mathematically or computationally. For this reason, the microscopic model like NETSIM can not be used to perform real-time traffic management of the highway network. However, some macroscopic models like SATURN assume traffic to be uniformly distributed during the modeled time interval and fails to capture essential features of traffic congestions [Ben Akiva et al. 1991]. To improve traffic modeling, an alternative approach turns to a combination of microscopic
and macroscopic modeling techniques, or a hybrid modeling approach. That is, we consider
the traffic stream as individual vehicles, but model their interactions macroscopically using
the average speed-density relationship. Our motivation is to use this kind of approach to
develop a traffic model that can be applied at the individual driver level and operated in a
real-time, or at least efficient, computing environment.

We describe the hybrid modeling approach by considering the three key issues of traffic
modeling: vehicle generation, network representation and vehicular movement.

3.2.1 Vehicle Generation

Vehicle generation describes how a traffic model generates or introduces simulated vehicle
arrival at a study network, or generates an arrival pattern. The arrival pattern can be mod-
eled by using the vehicle interarrival times (i.e., the times between two successive arrivals)
at each entry (source) to a network. These interarrival times are also called headways in
traffic simulation study. Headways at sources usually follow specific distributions, which
can be determined from observation and data analysis. A time-dependent arrival pattern
is required for a dynamic network flow modeling. That is, the arrival pattern should vary
with time.

For our present experiments, we simply use the negative exponential distribution to
represent the interarrival times. The probability density functions for this distribution is:

\[ f(t) = q e^{-qt} \]

Let \( F(t) = \int_0^t f(s)ds \), then \( F(t) = 1 - e^{-qt} \) or,

\[ t = -\frac{1}{q} \ln[1 - F(t)] \]

Hence, if the distribution of \( F(t) \) is uniform between zero and one, the above equation will
generate headways in traffic. However it is not completely satisfactory when dealing with
reasonably heavy traffic flows, as the negative exponential distribution tends to overesti-
mate the number of small headways (between zero and two seconds). To overcome this a
displaced exponential distribution is used [Young et al. 1989]. This distribution has no headways less than a certain minimum size, \( \tau \), called by the headway-threshold.

The probability of a headway greater than \( \tau \) is given by

\[
F(t) = 1 - \exp[-q(t - \tau)/(1 - q\tau)]
\]

or,

\[
t = \tau - \frac{1}{q} \ln[1 - F(t)]
\]

Thus, the arrival pattern at a specified source \( s \) is:

\[
t_n(s) = t_{n-1}(s) + \tau - \left( \frac{1}{q(s)} - \tau \right) \ln(R)
\]

where \( t_n(s) \) is the arrival time (second) of the \( n \)-th vehicle at source \( s \),

\( \tau \) is a constant and usually \( 0.0 < \tau \leq 2.0 \),

\( q(s) \) is the arrival rate (vehicles/minute/second) at source \( s \) and less than \( 1/\tau \), and

\( R \) is a random number between zero and one.

Arrival rate \( q(s) \) is usually computed from the statistics of traffic data at source \( s \) during some specified time interval. Let the total arrivals of vehicles during the measured period (\( T \)) be \( N(T) \), then \( q(s) = N(T)/T \).

However, as a module of our traffic models, vehicle generation is able to support a variety of traffic headway distributions at each source in the network. The module is also capable of running on trace data from traffic sensors. Sometimes, we can generate or prepare arrival times for all vehicles before the simulation and store these in a file. Then, the module of vehicle generation may read these data in the file.

### 3.2.2 A Flexible Network Representation

A highway network consists of a set of nodes and a set of links. The nodes are the intersections in the network where different traffic streams interact each other. The links are
the road sections between intersections. Traffic movement takes place along each link, and vehicles can only change their general direction of travel at the nodes. The characteristics of a route from an origin to a destination is the sum of the characteristics (e.g., travel time) of all the links that form the route. Users prefer a route that provides characteristics that satisfy some particular objectives (e.g., minimum travel time, or minimum cost).

A flexible network representation is necessary to accurately predict dynamic traffic characteristics. The network representation should be easily adapted and modified to better reflect the driver’s perception of the network. For this reason, we divide each link $L_i$ into a set of link segments $L_{i,j}$ by adding dummy nodes $T_{i,j}$ at the end of segment $L_{i,j}$, as shown in Figure 3.1. It is suggested that segment length not exceed 1 mile [Chang et al. 1985]. Each segment requires homogeneous geometry and similar physical features which may be represented by the data structure shown in Figure 3.1.

After the original network is divided, the “divided” network and the associated data
structures for these segments constitute a network representation. However, the dynamic nature of a traffic model requires that a network have different representations in peak hour as opposed to off-peak, on sunny days as opposed to rainy, on weekends as opposed to weekdays, in day as opposed to night, and so on. Therefore, a network representation should be associated with some time domain, say, \([a, b]\), where the length of \([a, b]\) reflects the dynamic nature of traffic flow. In addition, since each model has its own specific assumptions, different models may require different network representations.

In the following section, we will see this data structure of the link segment is well-suited to represent two modified Greenshields models.

### 3.2.3 Vehicular Movement: Two Assumptions

Giving a network representation during \([a, b]\), let \(Q_{i,j}(t)\) be the number of vehicles on segment \(L_{i,j}\) at time \(t \in [a, b]\). Since the mean density of a segment is the average number of vehicles per lane-mile in that segment, we may calculate the mean density \(K_{i,j}(t)\) of segment \(L_{i,j}\) at time \(t\) by the following formula:

\[
K_{i,j}(t) = \frac{Q_{i,j}(t)}{l_{i,j} \times \Delta L_{i,j}}
\]

To predict the dynamics of network traffic flow, we present two formal assumptions of vehicular movement. The first assumption is formulated according to the logic of MPSM [Mahmassani et al. 1988 and 1991]:

**Assumption 1** Given a network representation during \([a, b]\) and a fixed time-slice \((\Delta T)\), all vehicles on a segment \(L_{i,j}\) travel at the same mean speed during each \(\Delta T\), which is determined by the following equation:

\[
V_{i,j}(t) = V_{i,j}^d + (V_{i,j}^F - V_{i,j}^d)(1 - \frac{K_{i,j}(t)}{K_{i,j}^d})^{\alpha_{i,j}} 
\]

(3.2)

where \(V_{i,j}(t)\) is the mean speed for all vehicles in \(L_{i,j}\) during the \(t\)-th time-slice in \([a, b]\).
**Assumption 1** implies that vehicles change their (mean) speeds after each time-slice ($\Delta T$), and leads to an easy implementation for the time-driven simulation. However, as previously described, **Assumption 1** is not realistic in that the arrivals of newly entered vehicles in a segment affect the behaviors of vehicles that previously entered the segment, because **Assumption 1** implies that the mean speed of all vehicles on a segment slows down (speeds up) when the number of newly entered vehicles is more (less) than the number of the vehicles just exiting that segment. In addition, **Assumption 1** increases the sensitivity of the model’s accuracy to the selection of the time-slice, and leads to the constant state of the modeled quantities such as traffic flow and congestion during the given time-slice. Therefore, the time-slice length ($\Delta T$) specifies the dynamic resolution and limitation of the traffic model based on this assumption. We call equation 3.2 the **modified Greenshields model 1**.

To further improve traffic modeling, we present a new assumption:

**Assumption 2** Given a network representation during $[a, b]$, the average speed of each vehicle ($n$) on a segment ($L_{i,j}$) can be determined by the following equation:

$$V_{i,j}(n) = V_{i,j}^F + (V_{i,j}^F - V_{i,j}^J)(1 - \frac{K_{i,j}(t(n))}{K_{i,j}})\theta_{i,j} + \epsilon$$

(3.3)

where, $t(n)$ is the time when vehicle $n$ is entering $L_{i,j}$, $\epsilon$ is a constant, and $V_{i,j}(n)$ is the average speed of vehicle $n$ on $L_{i,j}$.

**Assumption 2** means that each vehicle determines its average speed on a segment when it enters that segment, and thus this average speed determines its travel time on the segment. That is, **Assumption 2** implies that each vehicle may change its average speed after each segment ($L_{i,j}$), and leads to an easy implementation for the event-driven simulation. Since different vehicles may have different speeds on a segment at any time, there is no limitation of dynamic resolution for the model caused by **Assumption 2**. Therefore, **Assumption 2** seems more realistic than **Assumption 1**, and provides a new theoretical tool for the development of dynamic traffic models.
We call equation 3.3 the modified Greenshields model II, by which the travel time \( t_{i,j}(n) \) for vehicle \( n \) on \( L_{i,j} \) can be easily obtained by \( \Delta L_{i,j}/V_{i,j}(n) \). In contrast, we cannot directly compute the travel time \( t_{i,j}(n) \) for vehicle \( n \) on \( L_{i,j} \) by equation 3.2. Instead, \( t_{i,j}(n) \) is obtained by adding all time-slices during which vehicle \( n \) traveled entirely on \( L_{i,j} \) plus the appropriate fractions of the time-slices when vehicle \( n \) entered and exited \( L_{i,j} \). Therefore, equation 3.3 can be used to compute the travel time for each vehicle on a segment more efficiently than equation 3.2.

For both assumptions, the travel time \( t_i(n) \) of vehicle \( n \) through link \( L_i \) is the sum of all travel times for \( n \) on the segments making up \( L_i \): \( \sum_j t_{i,j}(n) \). The total travel time \( t(n) \) of vehicle \( n \) is the sum of all travel times for \( n \) on the links making up its route: \( \sum_i t_i(n) \). The mean travel time for \( N \) vehicles on the same route is \( \sum_{n=1}^N t(n)/N \). When \( N \) is large enough, we refer to this mean travel time as the mean travel time of the route.

Notice that a traffic jam occurs in link \( L_i \) when \( K_{i,j}(t) \) for \( L_{i,j} \) (i.e., a segment of \( L_i \)) equals or exceeds the jam density \( K_{i,j}^j \). In this case, our implementation holds all vehicles that intend to enter \( L_{i,j} \) on a queue before \( L_{i,j} \) until \( K_{i,j}(t) \) falls below \( K_{i,j}^j \). A queue discipline is required, which describes how vehicles are to be selected for service from the pool of vehicles that have arrived at the queuing node. We use the standard First-In and First-Out principle to process vehicle arrivals on a node.

### 3.3 Summary

In this chapter, we evaluate the previous work in freeway traffic modeling. All existing freeway simulations use the time-driven approach. We describe a hybrid traffic modeling approach through vehicle generation, network representation, and vehicular movement. The hybrid modeling approach is a combination of microscopic traffic modeling techniques and macroscopic modeling logic of traffic flow. That is, the traffic stream is composed of individual vehicles, while the interactions in the traffic stream are modeled macroscopically using the average speed-density relationship, and microscopic details of car following, lane
changing, and signalized operation at junctions are not modeled.

A network representation depends on modeling requirements, along with physical road and traffic conditions. A proper representation will enhance the credibility of a traffic model. Since the speed-density parameters considerably affects the accuracy of simulation results, they should be estimated by considering weather conditions and vehicle types as well as physical road conditions.

The modified Greenshields model II makes it easy to compute the travel time of a vehicle on each segment and route. **Assumption 2** leads to an easy implementation of an event-driven simulation.
Chapter 4

IMPLEMENTATION OF TWO TRAFFIC MODELS

Chapter 1 introduced the concept of time-driven simulation and event-driven simulation. Chapter 3 described the two assumptions of vehicular movement. This chapter describes the implementation of a modified time-driven simulation and the event-driven simulation based on these assumptions.

4.1 A Modified Time-Driven Simulation

According to Assumption 1 and the time-driven approach, we have implemented a modified time-driven traffic simulation model (MTDS). The simulation begins with the first vehicle in the network at simulation clock, $t = 0$. The time clock is then advanced one time-slice to $\Delta T$, the positions of vehicles in the network are updated, and new states of each link and the network are determined. This process is repeated until the termination condition of the simulation is reached. Figure 4.1 show the simulation logic of MTDS. Here the major object of interest is the vehicle which has a data structure with it, as shown in Table 4.1. MTDS supports random arrival patterns and simulates vehicular movement one by one. MTDS loads each vehicle immediately when it arrives at a source. The mean
Table 4.1: The Data Structure of a Vehicle

<table>
<thead>
<tr>
<th>ATTRIBUTE</th>
<th>MEANING</th>
</tr>
</thead>
<tbody>
<tr>
<td>ArrivalTime</td>
<td>the time when the vehicle entered the network</td>
</tr>
<tr>
<td>VehType</td>
<td>the type of the vehicle</td>
</tr>
<tr>
<td>RouteNo</td>
<td>the number of the route the vehicle is taking</td>
</tr>
<tr>
<td>Position</td>
<td>the current position of the vehicle</td>
</tr>
<tr>
<td>TravelTimeOnL[i]</td>
<td>the travel time of the vehicle on Link[i]</td>
</tr>
</tbody>
</table>

speed of a new arrival, during the fraction of some time-slice, is dependent on the number of vehicles on its entering segment. Sometimes, we use MTDS-$\Delta T$ to represent the MTDS with the time-slice $\Delta T$ (seconds).

The major loop of MTDS-$\Delta T$ keeps track of the physical positions of vehicles by advancing the simulation clock in increments of $\Delta T$. Each route has an associated vehicle queue, which is sorted by the distances from the locations of vehicles to the destination along this route. A PositionUpdating routine is used to update the positions of a vehicle queue for each time-slice $\Delta T$. Let $P_{i,j}(n,t)$ denote the position of vehicle $n$ in $L_{i,j}$ at the end of the $t$-th time slice, and $R_{i,j}(n,t)$ the distance from its current position to the beginning of the next segment. Thus, $R_{i,j}(n,t) = \Delta L_{i,j} - P_{i,j}(n,t)$. Let $D(n,t+1) = \Delta T \times V_{i,j}(t+1)$, with two cases:

1. $D(n,t+1) \leq R_{i,j}(n,t)$. In this case, the new position of vehicle $n$ remains in $L_{i,j}$ at the end of the $(t+1)$-th time-slice, and is obtained by advancing $n$ by the distance $D(n,t+1)$.

2. $D(n,t+1) > R_{i,j}(n,t)$. In this case, vehicle $n$ must travel in the next segment, $L_{i',j'}$, during $\Delta T' = \Delta T - R_{i,j}(n,t)/V_{i,j}(t+1)$. Denote the distance that $n$ will travel during $\Delta T'$ by $R'_{i',j'}(n,t+1)$, as shown in Figure 4.2. That is, the new position of $n$ is obtained by advancing $n$ by the distance $R'_{i',j'}(n,t+1)$ on $L_{i',j'}$. There are the following two approaches to compute $R'_{i',j'}(n,t+1)$:

[A] Take $V_{i',j'}(t+1)$ as the mean speed for $n$ traveling on $L_{i',j'}$ during $\Delta T'$, then

$$R'_{i',j'}(n,t+1) = \Delta T' \times V_{i',j'}(t+1)$$
Figure 4.1: A Flowchart of MTDS
Figure 4.2: Vehicle Moving Process

Table 4.2: Physical Link Data

<table>
<thead>
<tr>
<th>LinkID</th>
<th>L1</th>
<th>L2</th>
<th>L3</th>
<th>L4</th>
<th>L5</th>
<th>L6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length (mile)</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Jam Density (veh/lane/mile)</td>
<td>170</td>
<td>180</td>
<td>200</td>
<td>190</td>
<td>185</td>
<td>185</td>
</tr>
<tr>
<td>Free Flow Speed (miles/hr)</td>
<td>55</td>
<td>60</td>
<td>65</td>
<td>65</td>
<td>55</td>
<td>50</td>
</tr>
<tr>
<td>Min Speed (miles/hr)</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>SD Parameter</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
</tr>
<tr>
<td>Num Lanes</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

[B] Take the average speed \( \frac{V_{i,j}(t+1) + V_{i',j'}(t+1)}{2} \) as the mean speed for \( n \) traveling on \( L_{i',j'} \) during \( \Delta T' \) [Chang et al. 1985], then

\[
R'_{i',j'}(n, t+1) = \frac{\Delta T'}{2}(V_{i,j}(t+1) + V_{i',j'}(t+1))
\]

These two approaches are called by Approach A and Approach B, respectively. We conduct a simple experiment to compare these two approaches. Figure 4.3 shows a section of freeway, with link data given in Table 4.2. Table 4.3 gives the simulated results of travel times, where the arrival rate at \( S \) is 0.25 veh/lane/sec. In MTDS-10 and MTDS-30, the travel times on \( L_3 \) and \( L_4 \) from Approach B make conflicts with the fact that the travel time on \( L_3 \) must be less than that on \( L_4 \) according to the physical data of \( L_3 \) and \( L_4 \). This

Figure 4.3: A Section of Freeway

31
Table 4.3: Comparison of Mean Travel Times (min)

| LINK ID | MTDS-1 | | MTDS-10 | | MTDS-30 | |
|---------|--------|--------|--------|--------|--------|
|         | A      | B      | A      | B      | A      | B      |
| $L_4$   | 1.238  | 1.238  | 1.238  | 1.238  | 1.238  | 1.238  |
| $L_5$   | 1.114  | 1.114  | 1.114  | 1.119  | 1.113  | 1.127  |
| $L_6$   | 1.009  | 1.009  | 1.008  | 1.013  | 1.007  | 1.020  |
| $L_7$   | 1.014  | 1.014  | 1.014  | 1.013  | 1.013  | 1.013  |
| $L_8$   | 1.224  | 1.223  | 1.223  | 1.214  | 1.221  | 1.197  |
| $L_9$   | 1.362  | 1.361  | 1.361  | 1.355  | 1.359  | 1.343  |

A simple experiment shows that Approach A is a little better than Approach B. Therefore, we choose Approach A to implement MTDS models.

4.2 An Event-Driven Simulation

According to Assumption 2 and the event-driven approach, we developed an event-driven traffic simulation (EDS) model. There are four basic types of events: $e_1$ reaches a source; $e_2$ reaches an intersection; $e_3$ reaches a dummy node; and $e_4$ departs from a destination. Since an event is invoked by some vehicle, the representation of the event includes information on the corresponding vehicle. The contents of the event consists of six elements: Occurrence-Time, EventType, NodeNo, ArrivalTime, VehType, and RouteNo, as shown in Table 4.4. That is, we can use $(t, T(e), N, t_0(n), T(n), R(n))$ to represent an event $e$ corresponding to vehicle $n$. For example, $(9.00, 2, T_1, 8.00, \text{car}, R_3)$ represents: "the car that entered the network at 8:00 AM is arriving at intersection $T_1$ at 9:00 AM along route $R_3$". These events are unconditional events that can be scheduled in advance. However, an incident is a conditional event that occurs somewhere and sometime in the network, and is specified by the time at which the set of incident boolean conditions becomes true.

The basic execution loop of the simulation repeatedly removes the smallest time-stamped event from the event set (queue) and processes that event. The event set is sorted by the
Table 4.4: The Data Structure of an Event

<table>
<thead>
<tr>
<th>ATTRIBUTE</th>
<th>MEANING</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NAME</strong></td>
<td><strong>SYMBOL</strong></td>
</tr>
<tr>
<td>OccurrenceTime</td>
<td>$t$</td>
</tr>
<tr>
<td>EventType</td>
<td>$T(e)$</td>
</tr>
<tr>
<td>NodeNo</td>
<td>$N$</td>
</tr>
<tr>
<td>ArrivalTime</td>
<td>$t_0(n)$</td>
</tr>
<tr>
<td>VehType</td>
<td>$T(n)$</td>
</tr>
<tr>
<td>RouteNo</td>
<td>$R(n)$</td>
</tr>
</tbody>
</table>

occurrence times of pending events. Each of four event types has its own processing routiae:

[Routine 1] Process event $e_1 = (t, 1, N, t_0(n), T(n), R(n))$ as follows:

1. Increment by one the number of vehicles in the segment that $n$ is entering.
2. Increment by one the number of vehicles in the network.
3. Check if the simulation has reached the steady state. If so, update the areas under the number of vehicles in the entering link versus time curve and under the number of vehicles in the network versus time curve, respectively.
4. Introduce a new vehicle to reach the same source and merge it into the event set.
5. Check the traffic congestion condition. If no congestion, update the speed for $n$, schedule a new arrival at next node along $R(n)$ and then merge it into the event set; if yes, set congestion flag to true and put vehicle $n$ in the waiting queue of origin $N$.
6. Delete $e_1$ from the event set.

[Routine 2] Process event $e_2 = (t, 2, N, t_0(n), T(n), R(n))$ as follows:

1. Subtract one from the number of vehicles in the segment where $n$ just exited.
2. Increment by one the number of vehicles in the segment vehicle $n$ is entering.
3. Increment by one the number of vehicles in the link containing the entering segment.
4. Check if the simulation has reached the steady state. If so, update the areas under the number of vehicles in the entering link versus time curve and under the number of vehicles in the exited link versus time curve, respectively.

33
5. Check the traffic congestion condition. If no congestion, update the speed for \( n \), schedule a new arrival at the next node along \( R(n) \) and then merge it into the event set; if yes, set congestion flag to true and put \( n \) in the waiting queue of node \( N \).

6. Delete \( e_2 \) from the event set.

7. If any vehicle reaches intersection \( N \) from outside of the network, go to **Routine 1**.

**[Routine 3]** Process event \( e_3 = (t, 3, N, t_0(n), T(n), R(n)) \) as follows:

1. Subtract one from the number of vehicles in the segment where \( n \) just exited.
2. Increment by one the number of vehicles in the segment vehicle \( n \) is entering.
3. Update the speed of vehicle \( n \).
4. Create a new arrival at the next node along \( R(n) \) and merge it into the event set
5. Delete \( e_3 \) from the event set.

**[Routine 4]** Process event \( e_4 = (t, 4, N, t_0(n), T(n), R(n)) \) as follows:

1. Subtract one from the number of vehicles in the segment where \( n \) just exited.
2. Subtract one from the number of vehicles in the link where \( n \) just exited.
3. Subtract one from the number of vehicles in the network.
4. Check if the simulation has reached the steady state. If so, update the areas under the number of vehicles in the exited link versus time curve and under the number of vehicles in the network versus time curve, respectively.
5. Delete \( e_4 \) from the event set.

Each routine tells what the simulation must do to process a particular event \( e \) if it is chosen for processing. The current implementation of the simulation terminates each run after the prespecified number of vehicles has been processed. The simulation logic of EDS is illustrated in Figure 4.4 [Balci 1988].

### 4.3 Computational Complexities of EDS and MTDS

Let \( M \) be the total number of nodes in the network representation and \( N \) be the number of vehicles to be simulated. The EDS model costs \( O(\sum_{i,j \in R_n} 1) \) to simulate a vehicle \( n \).
Figure 4.4: A Flowchart of EDS
through route $R_n$, and thus, the complexity of EDS is $O(NM)$. On the other hand, let the free travel time on $L_{i,j}$ be $T_{i,j}^F$, then $T_{i,j}^F = \Delta L_{i,j}/V_{i,j}^F$. The MTDS-$\Delta T$ model costs $O(\sum_{i,j\in R_n} (T_{i,j}^F/\Delta T))$ to simulate vehicle $n$ through route $R_n$, so the complexity of MTDS is $O(NMT_F/\Delta T)$, where $T_F$ is the average free travel time on all segments consisting of $R_n$, and usually $T_F \gg \Delta T$. Therefore, EDS is theoretically expected to be faster than MTDS-$\Delta T$ for a small $\Delta T$.

4.4 Summary

We develop an event-driven simulation model, which opens a new simulation environment for network traffic study. We compare the time-driven simulation models using a new processing approach versus the traditional one of vehicular movement over nodes during one time-slice. The results show that the new approach is a little better than the traditional one. We also give the theoretical analysis of the computational complexities of both event-driven simulations and time-driven simulations.
Chapter 5

EXPERIMENTAL COMPARISONS BETWEEN EDS AND MTDS

Chapter 4 implemented an event-driven traffic model (EDS) and a modified time-driven traffic model (MTDS). This chapter will focus on comparing the accuracy and efficiency of these two models based on three typical situations.

The objectives of this study are to:

1. investigate the effect of network configuration on traffic characteristics (i.e., travel time, traffic density and volume)
2. predict travel time and congestion situation for each link of a network as traffic flow changes dynamically at each source of the network
3. provide a tool to observe how the dynamics of route choice behavior affects the traffic density and travel time of the network
4. determine (if possible) which of the two simulation models is better in accuracy and efficiency for network traffic flow simulation
5.1 Simulation Experimentation

After the model has been conceptualized, coded, and tested, it is ready to provide information through experimentation.

The seeds used in random-number generation should not affect the final conclusion. Thus, the model should produce similar results for different seed values. It may be verified by running the simulation with different seed values. There are general guidelines for seed selection such as avoid even values, use nonoverlapping streams, and reuse seeds in successive replications [Law and Kelton 1991].

We design three typical scenarios to conduct the experiments to compare EDS and MTDS. Any real traffic system can be divided into a set of subsystems, each being one of these three typical situations. The programs are written in the C language. The execution times of the experiments are measured for one run on a DECstation 5000/133. The physical data for each link is selected to represent various traffic conditions. For present experiments, we set the headway-threshold as 1 second (see equation 3.1) at each source, that is, there is no headway less than 1 second. Both EDS and MTDS use the same link physical data. We also assume that all lanes of a link have the same arrival pattern. All steady-state performances in the following experiments are obtained by using the replication/deletion approach described in Section 2.5.

5.1.1 Scenario 1: A Single Highway

Figure 5.1 shows the network situation, a single freeway with one low-capacity link ($L_4$). The physical features for each link are given in Table 5.1.
Table 5.1: Physical Link Data in Scenario 1

<table>
<thead>
<tr>
<th>LinkID</th>
<th>$L_1$</th>
<th>$L_2$</th>
<th>$L_3$</th>
<th>$L_4$</th>
<th>$L_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length (mile)</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>JamDensity (veh/lane/mile)</td>
<td>170</td>
<td>180</td>
<td>200</td>
<td>120</td>
<td>185</td>
</tr>
<tr>
<td>FreeFlowSpeed (miles/hr)</td>
<td>55</td>
<td>60</td>
<td>65</td>
<td>45</td>
<td>55</td>
</tr>
<tr>
<td>MinSpeed (miles/hr)</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>SD_Parameter</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
</tr>
<tr>
<td>NumLanes</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

5.1.2 Warmup Periods

In order to perform a steady-state input, it is necessary to detect and eliminate warmup period. We made 20 independent replications of each simulation with length $m = 300 \times 100$ vehicles. Both EDS and MTDS use the same physical features for each link are given in Table 5.1. Figure 5.2a and Figure 5.2b plot the averaged trajectories of the total travel time from EDS and MTDS-1, respectively. The other runs for each model have similar trajectories to Figure 5.2. It is clear that further smoothing of the plot is necessary. Figure 5.3 plots the moving average trajectories of each model by Welch’s procedure with $50 \times 100$ window size. Here, $q(S) = .3$ vehicles per second as the arrival rate. The result shows that the warmup period is about 20,000 vehicles for both EDS and MTDS. Figure 5.4 analyzes the sensitivity of the warmup period to the traffic arrival rate, where the data comes from EDS. The result shows that the warmup periods for all arrival rates are almost the same, and thus needn’t be reset as the model simulates traffic flow dynamically.

Based on these experiments, we have the following results related to the warmup period:

1. The warmup period is about 20,000 vehicles for both EDS and MTDS.
2. The traffic arrival rate hardly affects the warmup period of the model.

In the following two scenarios, we conduct the similar experiments to analyze the warmup period. Those experiments also demonstrates the above two results. In the following experiments, therefore, we always set the warmup period to be 20,000 vehicles, and
(a) Averaged Total Travel Time from EDS

(b) Averaged Total Travel Time from MTDS-1

Figure 5.2: Comparison of Averaged Travel Time across Replications
Figure 5.3: Comparison of Warmup Period for Total Travel Time
Figure 5.4: Effect of Arrival Rate on Warmup Period
collected statistics for the next 200,000 vehicles.

We must claim that the warmup period will increase as the size of network becomes quite large. Generally, we set $\epsilon$ in equation 3.3 as 0.04, but may be slightly modified with different arrival rates.

5.1.3 Simulation Results

Table 5.2 and Table 5.3 give the comparisons of mean travel times and mean traffic densities, respectively. They demonstrate that (1) MTDS-\(\Delta T\) converges to MTDS-1 when \(\Delta T\) approaches to 1 second; (2) MTDS-9.2 has the similar results to MTDS-1; (3) EDS produces results similar to MTDS-1; and (4) the differences between MTDS-1, MTDS-10 and MTDS-30 becomes larger when the arrival rate gets bigger. Why is 1 second so important? The reason is that we took 1 second as the headway-threshold in vehicle generation. That is, MTDS can process at most one vehicle during each 1 second period. From Table 5.3, we also notice that the big difference of traffic density lies in the first link. This is due to Assumption 1 which assumes link state is constant and not affected by new arrivals during each time-slice.

Little’s Law in Section 2.6 is useful to test the validity of the simulation model. In Table 5.4, theoretical volume = arrival rate \times an hour, where the arrival rate is assumed constant during the hour. The simulated volume is computed by using the fundamental equation 2.2 of traffic characteristics, where float is truncated into the next lower integer. Table 5.4 shows that EDS produces the same volume to the theoretical one. The error of MTDS-\(\Delta T\) results from Assumption 1 and thus, the bigger \(\Delta T\) results in the bigger error. In this sense, the EDS model is more accurate than the MTDS model, and MTDS-\(\Delta T\) gets more accurate when \(\Delta T\) approaches to zero. Table 5.4 also shows that when traffic flow is greater than 0.35 veh/lane/sec, there happens the serious traffic jam in the highway. This is caused by the low capacity of \(L_4\). Table 5.5 shows that the speedup of EDS is about 4 times to MTDS-1.
Table 5.2: Comparison of Mean Travel Times (min) of Scenario 1

<table>
<thead>
<tr>
<th>LINK ID</th>
<th>EDS</th>
<th>MTDS-0.2</th>
<th>MTDS-1</th>
<th>MTDS-10</th>
<th>MTDS-30</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$q(S) = 0.15$ (veh/lane/sec)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$L_1$</td>
<td>1.174</td>
<td>1.174</td>
<td>1.174</td>
<td>1.174</td>
<td>1.173</td>
</tr>
<tr>
<td>$L_2$</td>
<td>1.065</td>
<td>1.066</td>
<td>1.066</td>
<td>1.065</td>
<td>1.064</td>
</tr>
<tr>
<td>$L_3$</td>
<td>0.972</td>
<td>0.973</td>
<td>0.973</td>
<td>0.973</td>
<td>0.972</td>
</tr>
<tr>
<td>$L_4$</td>
<td>1.525</td>
<td>1.522</td>
<td>1.522</td>
<td>1.521</td>
<td>1.519</td>
</tr>
<tr>
<td>$L_5$</td>
<td>1.165</td>
<td>1.166</td>
<td>1.166</td>
<td>1.165</td>
<td>1.164</td>
</tr>
<tr>
<td>Total</td>
<td>5.901</td>
<td>5.901</td>
<td>5.901</td>
<td>5.898</td>
<td>5.892</td>
</tr>
<tr>
<td></td>
<td>$q(S) = 0.25$ (veh/lane/sec)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$L_1$</td>
<td>1.239</td>
<td>1.239</td>
<td>1.239</td>
<td>1.238</td>
<td>1.238</td>
</tr>
<tr>
<td>$L_2$</td>
<td>1.113</td>
<td>1.115</td>
<td>1.115</td>
<td>1.114</td>
<td>1.113</td>
</tr>
<tr>
<td>$L_3$</td>
<td>1.008</td>
<td>1.009</td>
<td>1.009</td>
<td>1.008</td>
<td>1.007</td>
</tr>
<tr>
<td>$L_4$</td>
<td>1.722</td>
<td>1.719</td>
<td>1.718</td>
<td>1.717</td>
<td>1.714</td>
</tr>
<tr>
<td>$L_5$</td>
<td>1.222</td>
<td>1.223</td>
<td>1.223</td>
<td>1.222</td>
<td>1.221</td>
</tr>
<tr>
<td></td>
<td>$q(S) = 0.3$ (veh/lane/sec)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$L_1$</td>
<td>1.277</td>
<td>1.277</td>
<td>1.277</td>
<td>1.277</td>
<td>1.276</td>
</tr>
<tr>
<td>$L_2$</td>
<td>1.142</td>
<td>1.143</td>
<td>1.143</td>
<td>1.142</td>
<td>1.142</td>
</tr>
<tr>
<td>$L_3$</td>
<td>1.028</td>
<td>1.029</td>
<td>1.029</td>
<td>1.028</td>
<td>1.027</td>
</tr>
<tr>
<td>$L_4$</td>
<td>1.881</td>
<td>1.879</td>
<td>1.878</td>
<td>1.875</td>
<td>1.870</td>
</tr>
<tr>
<td>$L_5$</td>
<td>1.255</td>
<td>1.257</td>
<td>1.257</td>
<td>1.256</td>
<td>1.254</td>
</tr>
<tr>
<td>Total</td>
<td>6.583</td>
<td>6.585</td>
<td>6.584</td>
<td>6.578</td>
<td>6.569</td>
</tr>
</tbody>
</table>
Table 5.3: Comparison of Mean Densities (veh/lane/mile) of Scenario 1

<table>
<thead>
<tr>
<th>LINK ID</th>
<th>EDS</th>
<th>MTDS-0.2</th>
<th>MTDS-1</th>
<th>MTDS-10</th>
<th>MTDS-30</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$L_1$</td>
<td>10.571</td>
<td>10.541</td>
<td>10.410</td>
<td>9.059</td>
<td>6.059</td>
</tr>
<tr>
<td>$L_3$</td>
<td>8.751</td>
<td>8.754</td>
<td>8.754</td>
<td>8.750</td>
<td>8.338</td>
</tr>
<tr>
<td>$L_5$</td>
<td>10.493</td>
<td>10.492</td>
<td>10.491</td>
<td>10.483</td>
<td>10.474</td>
</tr>
<tr>
<td>Total</td>
<td>10.628</td>
<td>10.590</td>
<td>10.587</td>
<td>10.311</td>
<td>9.703</td>
</tr>
</tbody>
</table>

$q(S) = 0.15$ (veh/lane/sec)

<table>
<thead>
<tr>
<th>LINK ID</th>
<th>EDS</th>
<th>MTDS-0.2</th>
<th>MTDS-1</th>
<th>MTDS-10</th>
<th>MTDS-30</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$L_1$</td>
<td>18.584</td>
<td>18.535</td>
<td>18.324</td>
<td>16.072</td>
<td>11.059</td>
</tr>
<tr>
<td>$L_3$</td>
<td>15.112</td>
<td>15.125</td>
<td>15.125</td>
<td>15.118</td>
<td>15.108</td>
</tr>
<tr>
<td>$L_5$</td>
<td>18.335</td>
<td>18.334</td>
<td>18.334</td>
<td>18.330</td>
<td>18.302</td>
</tr>
<tr>
<td>Total</td>
<td>18.915</td>
<td>18.882</td>
<td>18.857</td>
<td>18.395</td>
<td>17.373</td>
</tr>
</tbody>
</table>

$q(S) = 0.25$ (veh/lane/sec)

<table>
<thead>
<tr>
<th>LINK ID</th>
<th>EDS</th>
<th>MTDS-0.2</th>
<th>MTDS-1</th>
<th>MTDS-10</th>
<th>MTDS-30</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$L_1$</td>
<td>22.999</td>
<td>22.887</td>
<td>22.686</td>
<td>19.983</td>
<td>13.969</td>
</tr>
<tr>
<td>$L_2$</td>
<td>20.558</td>
<td>20.569</td>
<td>20.567</td>
<td>20.559</td>
<td>20.542</td>
</tr>
<tr>
<td>$L_3$</td>
<td>18.493</td>
<td>18.510</td>
<td>18.509</td>
<td>18.506</td>
<td>18.499</td>
</tr>
<tr>
<td>$L_4$</td>
<td>33.875</td>
<td>33.789</td>
<td>33.789</td>
<td>33.736</td>
<td>33.640</td>
</tr>
<tr>
<td>$L_5$</td>
<td>22.601</td>
<td>22.615</td>
<td>22.615</td>
<td>22.595</td>
<td>22.587</td>
</tr>
<tr>
<td>Total</td>
<td>23.705</td>
<td>23.660</td>
<td>23.634</td>
<td>23.076</td>
<td>21.848</td>
</tr>
</tbody>
</table>

$q(S) = 0.3$ (veh/lane/sec)
Table 5.4: Comparison of Effect of Arrival Rate on Major Performances of Scenario 1

<table>
<thead>
<tr>
<th>ARRIVAL RATE (veh/lane/sec)</th>
<th>EDS</th>
<th>MTDS-0.2</th>
<th>MTDS-1</th>
<th>MTDS-10</th>
<th>MTDS-30</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.15</td>
<td>5.901</td>
<td>5.901</td>
<td>5.901</td>
<td>5.898</td>
<td>5.892</td>
</tr>
<tr>
<td>0.20</td>
<td>6.084</td>
<td>6.085</td>
<td>6.085</td>
<td>6.082</td>
<td>6.075</td>
</tr>
<tr>
<td>0.25</td>
<td>6.304</td>
<td>6.305</td>
<td>6.304</td>
<td>6.299</td>
<td>6.293</td>
</tr>
<tr>
<td>0.30</td>
<td>6.583</td>
<td>6.585</td>
<td>6.584</td>
<td>6.578</td>
<td>6.569</td>
</tr>
<tr>
<td>0.36</td>
<td>Jam</td>
<td>Jam</td>
<td>Jam</td>
<td>Jam</td>
<td>Jam</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TRAFFIC DENSITY (veh/lane/mile)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.15</td>
</tr>
<tr>
<td>0.25</td>
</tr>
<tr>
<td>0.30</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>THEORETICAL VOLUME (veh/lane/hr)</th>
<th>SIMULATED VOLUME (veh/lane/hr)</th>
</tr>
</thead>
<tbody>
<tr>
<td>540</td>
<td>540</td>
</tr>
<tr>
<td>720</td>
<td>720</td>
</tr>
<tr>
<td>900</td>
<td>900</td>
</tr>
<tr>
<td>1080</td>
<td>1080</td>
</tr>
</tbody>
</table>

Table 5.5: Comparison of Execution Times of Scenario 1

<table>
<thead>
<tr>
<th>MODEL</th>
<th>EDS</th>
<th>MTDS-0.2</th>
<th>MTDS-1</th>
<th>MTDS-10</th>
<th>MTDS-30</th>
</tr>
</thead>
<tbody>
<tr>
<td>Execution Time (sec)</td>
<td>85</td>
<td>1770</td>
<td>350</td>
<td>50</td>
<td>22</td>
</tr>
</tbody>
</table>
Figure 5.5: Scenario 2 — a Freeway with two Ramp Junctions

<table>
<thead>
<tr>
<th>LinkID</th>
<th>$L_1$</th>
<th>$L_2$</th>
<th>$L_3$</th>
<th>$L_4$</th>
<th>$L_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length (mile)</td>
<td>1</td>
<td>1</td>
<td>i</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>JamDensity (veh/lane/mile)</td>
<td>190</td>
<td>180</td>
<td>210</td>
<td>200</td>
<td>190</td>
</tr>
<tr>
<td>FreeFlowSpeed (miles/hr)</td>
<td>50</td>
<td>65</td>
<td>55</td>
<td>55</td>
<td>55</td>
</tr>
<tr>
<td>MinSpeed (miles/hr)</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>SD Parameter</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
</tr>
<tr>
<td>NumLanes</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

5.2 Scenario 2: A Merging Highway

Figure 5.5 shows the network situation that consists of a source ($S$), an on-ramp point ($T_3$), an off-ramp point ($T_4$), a destination ($D$) and five links $L_1$, $L_2$, $L_3$, $L_4$ and $L_5$. Table 5.6 shows the physical features for each link. An on-ramp provides for the merging of the entering vehicles with a shoulder lane of through traffic on the freeway. An off-ramp allows vehicles to diverge from a freeway lane onto an existing roadway. Ramp junctions are those short segments of freeway where vehicles are permitted to enter or leave the freeway.

Table 5.7 compares the major performances between EDS, MTDS-0.2, MTDS-1, MTDS-10, and MTDS-30. Like the first scenario, the results show that (1) MTDS-$\Delta T$ converges to MTDS-1 when $\Delta T$ approaches to 1 second; (2) there is little difference between MTDS-0.2 and MTDS-1; (3) for MTDS-$\Delta T$, the major error of volume that lies in the first link gets smaller when $\Delta T$ approaches to zero; and (4) EDS produces results similar to MTDS-0.2 and MTDS-1. However, EDS speeds up about 4 times to MTDS-1, as shown in Table 5.8.
Table 5.7: Comparison of Major Performances of Scenario 2

<table>
<thead>
<tr>
<th>q(S)</th>
<th>q(T3)</th>
<th>q(T4)</th>
<th>LINK</th>
<th>EDS</th>
<th>MTDS-0.2</th>
<th>MTDS-1</th>
<th>MTDS-10</th>
<th>MTDS-30</th>
</tr>
</thead>
<tbody>
<tr>
<td>(veh/sec)</td>
<td>ID</td>
<td>MEAN TRAVEL TIME (min)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.4</td>
<td>0.2</td>
<td>0.1</td>
<td>L1</td>
<td>1.32</td>
<td>1.32</td>
<td>1.32</td>
<td>1.32</td>
<td>1.319</td>
</tr>
<tr>
<td></td>
<td>L2</td>
<td>0.997</td>
<td>0.999</td>
<td>0.999</td>
<td>0.999</td>
<td>0.998</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L3</td>
<td>1.118</td>
<td>1.118</td>
<td>1.118</td>
<td>1.117</td>
<td>1.116</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L4</td>
<td>1.243</td>
<td>1.241</td>
<td>1.241</td>
<td>1.240</td>
<td>1.239</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L5</td>
<td>1.013</td>
<td>1.014</td>
<td>1.014</td>
<td>1.013</td>
<td>1.012</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td></td>
<td>5.691</td>
<td>5.692</td>
<td>5.692</td>
<td>5.689</td>
<td>5.684</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MEAN TRAFFIC DENSITY (veh/lane/mile)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L1</td>
<td>15.84</td>
<td>15.82</td>
<td>15.65</td>
<td>13.83</td>
<td>9.82</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L2</td>
<td>11.97</td>
<td>11.99</td>
<td>11.99</td>
<td>11.97</td>
<td>11.95</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L3</td>
<td>20.14</td>
<td>20.12</td>
<td>20.11</td>
<td>20.09</td>
<td>20.07</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L4</td>
<td>22.38</td>
<td>22.34</td>
<td>22.34</td>
<td>22.31</td>
<td>22.26</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L5</td>
<td>15.20</td>
<td>15.20</td>
<td>15.20</td>
<td>15.19</td>
<td>15.17</td>
<td></td>
<td></td>
</tr>
<tr>
<td>THEORETICAL VOLUME (veh/lane/hr)</td>
<td></td>
<td></td>
<td></td>
<td>SIMULATED VOLUME (veh/lane/hr)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>720</td>
<td>L1</td>
<td>720</td>
<td>719</td>
<td>711</td>
<td>628</td>
<td>446</td>
<td></td>
<td></td>
</tr>
<tr>
<td>720</td>
<td>L2</td>
<td>720</td>
<td>720</td>
<td>720</td>
<td>718</td>
<td>718</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1080</td>
<td>L3</td>
<td>1080</td>
<td>1079</td>
<td>1079</td>
<td>1079</td>
<td>1079</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1080</td>
<td>L4</td>
<td>1080</td>
<td>1080</td>
<td>1080</td>
<td>1079</td>
<td>1077</td>
<td></td>
<td></td>
</tr>
<tr>
<td>900</td>
<td>L5</td>
<td>900</td>
<td>899</td>
<td>899</td>
<td>899</td>
<td>899</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MEAN TRAVEL TIME (min)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L1</td>
<td>1.397</td>
<td>1.397</td>
<td>1.397</td>
<td>1.396</td>
<td>1.397</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L2</td>
<td>1.042</td>
<td>1.044</td>
<td>1.044</td>
<td>1.043</td>
<td>1.042</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L3</td>
<td>1.170</td>
<td>1.169</td>
<td>1.169</td>
<td>1.168</td>
<td>1.167</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L4</td>
<td>1.313</td>
<td>1.311</td>
<td>1.311</td>
<td>1.310</td>
<td>1.308</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L5</td>
<td>1.058</td>
<td>1.059</td>
<td>1.059</td>
<td>1.058</td>
<td>1.057</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>5.980</td>
<td>5.980</td>
<td>5.980</td>
<td>5.975</td>
<td>5.969</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>MEAN TRAFFIC DENSITY (veh/lane/mile)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L1</td>
<td>25.16</td>
<td>25.10</td>
<td>24.85</td>
<td>22.13</td>
<td>16.11</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L2</td>
<td>18.77</td>
<td>18.79</td>
<td>18.79</td>
<td>18.77</td>
<td>18.75</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L3</td>
<td>28.08</td>
<td>28.04</td>
<td>28.04</td>
<td>28.02</td>
<td>28.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L4</td>
<td>31.52</td>
<td>31.46</td>
<td>31.44</td>
<td>31.41</td>
<td>31.37</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>L5</td>
<td>22.23</td>
<td>22.25</td>
<td>22.25</td>
<td>22.22</td>
<td>22.20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>THEORETICAL VOLUME (veh/lane/hr)</td>
<td></td>
<td></td>
<td></td>
<td>SIMULATED VOLUME (veh/lane/hr)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1080</td>
<td>L1</td>
<td>1080</td>
<td>1077</td>
<td>1067</td>
<td>951</td>
<td>692</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1080</td>
<td>L2</td>
<td>1080</td>
<td>1079</td>
<td>1079</td>
<td>1079</td>
<td>1079</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1440</td>
<td>L3</td>
<td>1440</td>
<td>1439</td>
<td>1439</td>
<td>1439</td>
<td>1439</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1440</td>
<td>L4</td>
<td>1440</td>
<td>1439</td>
<td>1438</td>
<td>1438</td>
<td>1438</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1260</td>
<td>L5</td>
<td>1260</td>
<td>1260</td>
<td>1260</td>
<td>1260</td>
<td>1260</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 5.8: Comparison of Execution Times of Scenario 2

<table>
<thead>
<tr>
<th>MODEL</th>
<th>EDS</th>
<th>MTDS-0.2</th>
<th>MTDS-1</th>
<th>MTDS-10</th>
<th>MTDS-30</th>
</tr>
</thead>
<tbody>
<tr>
<td>Execution</td>
<td>64</td>
<td>1360</td>
<td>258</td>
<td>43</td>
<td>19</td>
</tr>
<tr>
<td>Time (sec)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 5.6: Scenario 3 — a Sample Network

5.3 Scenario 3: A Sample Network

The sample network is illustrated in Figure 5.6, with link data given in Table 5.9. There are eight different routes, as shown in Table 5.10. For simplicity, we assume that each vehicle enters the network with a predetermined route. We associate one random generator with one route to generate the arrival pattern of that route.

Table 5.9: Physical Link Data in Scenario 3

<table>
<thead>
<tr>
<th>LinkID</th>
<th>( L_1 )</th>
<th>( L_2 )</th>
<th>( L_3 )</th>
<th>( L_4 )</th>
<th>( L_5 )</th>
<th>( L_6 )</th>
<th>( L_7 )</th>
<th>( L_8 )</th>
<th>( L_9 )</th>
<th>( L_{10} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length (mile)</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>JamDensity (veh/lane/mile)</td>
<td>180</td>
<td>180</td>
<td>200</td>
<td>200</td>
<td>180</td>
<td>180</td>
<td>180</td>
<td>180</td>
<td>180</td>
<td>180</td>
</tr>
<tr>
<td>FreeFlowSpeed (mile/hr)</td>
<td>45</td>
<td>65</td>
<td>65</td>
<td>65</td>
<td>55</td>
<td>55</td>
<td>45</td>
<td>55</td>
<td>65</td>
<td>45</td>
</tr>
<tr>
<td>MinSpeed (mile/hr)</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>SD_Parameter</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
<td>1.2</td>
</tr>
<tr>
<td>NumLanes</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>
Table 5.10: Eight Routes in Scenario 3

<table>
<thead>
<tr>
<th>ROUTE NO.</th>
<th>ROUTE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$S_1 \rightarrow T_1 \rightarrow T_3 \rightarrow D_1$</td>
</tr>
<tr>
<td>2</td>
<td>$S_1 \rightarrow T_1 \rightarrow T_2 \rightarrow T_4 \rightarrow T_3 \rightarrow D_1$</td>
</tr>
<tr>
<td>3</td>
<td>$S_1 \rightarrow T_1 \rightarrow T_2 \rightarrow T_4 \rightarrow D_2$</td>
</tr>
<tr>
<td>4</td>
<td>$S_1 \rightarrow T_1 \rightarrow T_3 \rightarrow T_4 \rightarrow D_2$</td>
</tr>
<tr>
<td>5</td>
<td>$S_2 \rightarrow T_2 \rightarrow T_4 \rightarrow D_2$</td>
</tr>
<tr>
<td>6</td>
<td>$S_2 \rightarrow T_2 \rightarrow T_1 \rightarrow T_3 \rightarrow T_4 \rightarrow D_2$</td>
</tr>
<tr>
<td>7</td>
<td>$S_2 \rightarrow T_2 \rightarrow T_1 \rightarrow T_3 \rightarrow D_1$</td>
</tr>
<tr>
<td>8</td>
<td>$S_2 \rightarrow T_2 \rightarrow T_4 \rightarrow T_3 \rightarrow D_1$</td>
</tr>
</tbody>
</table>

Table 5.11 assumes that all arrival rates $q(R_i)$ of route $R_i$ ($i = 1, 2, ..., 8$) are 0.1 veh/lane/sec. The results agree to these in the first two scenarios. EDS produces results similar to MTDS-0.2 and MTDS-1. Since the arrival rate for each route is quite low (i.e., 1 vehicle per 10 seconds), the difference between MTDS-1, MTDS-10 and MTDS-30 is little. Table 5.13 shows that EDS speeds up 2.6 times to MTDS-1. Compared the first scenario, the speedups of EDS to MTDS-1 and MTDS-0.2 are decreased. The reason for this is that the size of the global event queue in EDS is increased, but the size of each route queue in MTDS keeps small due to optimally allocating vehicles into eight different queues associated with their routes. However, we can also speedup EDS for large network applications by using some optimal event-set algorithms such as calendar queues [Brown 1988].

5.4 Conclusions

We conduct three experiments to compare the event-driven traffic simulation model versus the time-driven traffic simulation model in both accuracy and efficiency. Both EDS and MTDS are general-purpose freeway simulation models, not limited the above three situations. These models can be further developed into on-line application software packages to inform the driver about current and near-future traffic conditions based on the dynamics
Table 5.11: Comparison of Major Performances of Scenario 3

<table>
<thead>
<tr>
<th>LINK ID</th>
<th>EDS</th>
<th>MTDS-0.2</th>
<th>MTDS-1</th>
<th>MTDS-16</th>
<th>MTDS-30</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MEAN TRAVEL TIME (min) ON EACH LINK</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L1</td>
<td>1.772</td>
<td>1.77</td>
<td>1.77</td>
<td>1.764</td>
<td>1.76</td>
</tr>
<tr>
<td>L2</td>
<td>1.104</td>
<td>1.102</td>
<td>1.102</td>
<td>1.10</td>
<td></td>
</tr>
<tr>
<td>L3</td>
<td>1.078</td>
<td>1.078</td>
<td>1.078</td>
<td>1.078</td>
<td>1.076</td>
</tr>
<tr>
<td>L4</td>
<td>1.078</td>
<td>1.078</td>
<td>1.078</td>
<td>1.066</td>
<td>1.06</td>
</tr>
<tr>
<td>L5</td>
<td>1.356</td>
<td>1.358</td>
<td>1.358</td>
<td>1.356</td>
<td>1.352</td>
</tr>
<tr>
<td>L6</td>
<td>1.356</td>
<td>1.358</td>
<td>1.358</td>
<td>1.354</td>
<td>1.35</td>
</tr>
<tr>
<td>L7</td>
<td>1.492</td>
<td>1.492</td>
<td>1.492</td>
<td>1.49</td>
<td>1.49</td>
</tr>
<tr>
<td>L8</td>
<td>1.20</td>
<td>1.20</td>
<td>1.20</td>
<td>1.20</td>
<td>1.20</td>
</tr>
<tr>
<td>L9</td>
<td>0.998</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>0.998</td>
</tr>
<tr>
<td>L10</td>
<td>1.49</td>
<td>1.49</td>
<td>1.49</td>
<td>1.49</td>
<td>1.49</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>MEAN DENSITY (veh/lane/mile) ON EACH LINK</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>L1</td>
<td>42.54</td>
<td>42.54</td>
<td>42.54</td>
<td>42.42</td>
<td>42.40</td>
</tr>
<tr>
<td>L2</td>
<td>26.50</td>
<td>26.50</td>
<td>26.50</td>
<td>26.48</td>
<td>26.48</td>
</tr>
<tr>
<td>L3</td>
<td>25.86</td>
<td>25.86</td>
<td>25.86</td>
<td>25.84</td>
<td>25.84</td>
</tr>
<tr>
<td>L4</td>
<td>25.86</td>
<td>25.86</td>
<td>25.86</td>
<td>25.54</td>
<td>25.42</td>
</tr>
<tr>
<td>L5</td>
<td>32.56</td>
<td>32.56</td>
<td>32.56</td>
<td>32.52</td>
<td>32.36</td>
</tr>
<tr>
<td>L6</td>
<td>32.56</td>
<td>32.56</td>
<td>32.56</td>
<td>32.52</td>
<td>32.36</td>
</tr>
<tr>
<td>L7</td>
<td>17.92</td>
<td>17.90</td>
<td>17.88</td>
<td>17.86</td>
<td>17.86</td>
</tr>
<tr>
<td>L9</td>
<td>11.98</td>
<td>12.04</td>
<td>12.04</td>
<td>12.04</td>
<td>11.96</td>
</tr>
<tr>
<td>L10</td>
<td>17.88</td>
<td>17.88</td>
<td>17.88</td>
<td>17.86</td>
<td>17.84</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ROUTE NO.</th>
<th>EDS</th>
<th>MTDS-0.2</th>
<th>MTDS-1</th>
<th>MTDS-10</th>
<th>MTDS-30</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MEAN TRAVEL TIME (min) ON EACH ROUTE</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>4.206</td>
<td>4.206</td>
<td>4.206</td>
<td>4.198</td>
<td>4.188</td>
</tr>
<tr>
<td>3</td>
<td>5.406</td>
<td>5.406</td>
<td>5.406</td>
<td>5.384</td>
<td>5.37</td>
</tr>
<tr>
<td>4</td>
<td>5.696</td>
<td>5.696</td>
<td>5.696</td>
<td>5.686</td>
<td>5.676</td>
</tr>
<tr>
<td>5</td>
<td>3.538</td>
<td>3.538</td>
<td>3.538</td>
<td>3.52</td>
<td>3.51</td>
</tr>
<tr>
<td>6</td>
<td>6.52</td>
<td>6.52</td>
<td>6.52</td>
<td>6.512</td>
<td>6.506</td>
</tr>
<tr>
<td>7</td>
<td>5.03</td>
<td>5.03</td>
<td>5.03</td>
<td>5.024</td>
<td>5.018</td>
</tr>
<tr>
<td>8</td>
<td>4.536</td>
<td>4.538</td>
<td>4.538</td>
<td>4.522</td>
<td>4.51</td>
</tr>
</tbody>
</table>
Table 5.12: Comparison of Traffic Volume of Scenario 3

<table>
<thead>
<tr>
<th>LINK ID</th>
<th>THEORETICAL VOLUME (veh/lane/hr)</th>
<th>EDS</th>
<th>MTDS-0.2</th>
<th>MTDS-1</th>
<th>MTDS-10</th>
<th>MTDS-30</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L_1$</td>
<td>1440</td>
<td>1440</td>
<td>1442</td>
<td>1442</td>
<td>1442</td>
<td>1445</td>
</tr>
<tr>
<td>$L_2$</td>
<td>1440</td>
<td>1440</td>
<td>1442</td>
<td>1442</td>
<td>1444</td>
<td>1444</td>
</tr>
<tr>
<td>$L_3$</td>
<td>1440</td>
<td>1439</td>
<td>1439</td>
<td>1439</td>
<td>1439</td>
<td>1338</td>
</tr>
<tr>
<td>$L_4$</td>
<td>1440</td>
<td>1439</td>
<td>1439</td>
<td>1439</td>
<td>1442</td>
<td>1338</td>
</tr>
<tr>
<td>$L_5$</td>
<td>1440</td>
<td>1440</td>
<td>1438</td>
<td>1438</td>
<td>1438</td>
<td>1436</td>
</tr>
<tr>
<td>$L_6$</td>
<td>1440</td>
<td>1440</td>
<td>1438</td>
<td>1438</td>
<td>1441</td>
<td>1438</td>
</tr>
<tr>
<td>$L_7$</td>
<td>720</td>
<td>720</td>
<td>720</td>
<td>719</td>
<td>719</td>
<td>719</td>
</tr>
<tr>
<td>$L_8$</td>
<td>720</td>
<td>719</td>
<td>719</td>
<td>719</td>
<td>719</td>
<td>716</td>
</tr>
<tr>
<td>$L_9$</td>
<td>720</td>
<td>720</td>
<td>722</td>
<td>722</td>
<td>722</td>
<td>719</td>
</tr>
<tr>
<td>$L_{10}$</td>
<td>720</td>
<td>719</td>
<td>719</td>
<td>719</td>
<td>719</td>
<td>718</td>
</tr>
</tbody>
</table>

Table 5.13: Comparison of Execution Times of Scenario 3

<table>
<thead>
<tr>
<th>MODEL</th>
<th>EDS</th>
<th>MTDS-0.2</th>
<th>MTDS-1</th>
<th>MTDS-10</th>
<th>MTDS-30</th>
</tr>
</thead>
<tbody>
<tr>
<td>Execution Time (sec)</td>
<td>160</td>
<td>2080</td>
<td>420</td>
<td>55</td>
<td>26</td>
</tr>
</tbody>
</table>
of the traffic flow on entry to the network.

Generally, we conclude as follows:

1. MTDS-$\Delta T$ converges to MTDS-$\tau$ when $\Delta T$ approaches to $\tau$, and MTDS-$\Delta T$ behaves roughly like MTDS-$\tau$ when $\Delta T$ gets smaller than $\tau$, where $\tau$ is the headway-threshold.

2. EDS produces results similar to MTDS-$\tau$, and is more efficient than MTDS-$\tau$, where $\tau$ is the headway-threshold and usually less than 2 seconds [Young et al. 1989].

3. Compared to the theoretical volume, EDS is very accurate and MTDS-$\Delta T$ becomes more accurate when $\Delta T$ gets smaller.

4. The differences between MTDS-1, MTDS-10 and MTDS-30 depends on the arrival rates. When the arrival rate is quite small, we may choose ten seconds or even bigger as the time-slice for MTDS for the better trade-off between efficiency and accuracy.

We believe that the event-driven freeway simulation offers an efficient alternative to the traditional time-driven freeway simulation.
Chapter 6

CONCLUSIONS AND FUTURE WORK

Increasing levels of traffic congestion underscore the importance of the Advanced Traffic Management System (ATMS). One major goal of ATMS is to improve highway systems performance by providing real-time route guidance information to on-road drivers. This goal makes it necessary for traffic models to capture the dynamic nature of traffic flow through the network and to predict the travel time for each link and each route. This project has made some progress toward the goal. The project has addressed several key issues of the development of traffic simulation such as how to generate vehicle arrivals, how to represent a transportation network, how to investigate vehicle movement, how to test a traffic simulation model, and how to obtain the desired statistical analysis of the simulation outputs. A subjective review of the widely publicized traffic simulation models is considered. A presentation and analysis of the experimental results is included.

Although past research efforts have produced some freeway simulation models, this research accomplishes four major tasks not achieved by other efforts:

1. Presents a hybrid traffic modeling approach for the development of traffic flow simulations within a reasonable time-scale.

2. Develops a general-purpose traffic flow simulation model by using the event-driven
approach, which provides a new simulation tool for freeway traffic study.

3. Develops a time-driven traffic simulation model that supports random arrival patterns. An interesting result is that all MTDS-$\Delta T$ converges to MTDS-$\tau$, where $\tau$ is the headway-threshold.

4. Conducts a comprehensive comparison between EDS and MTDS. The experimental results show that the event-driven simulation is competitive in efficiency and accuracy with traditional time-driven simulation using small time slices.

The hybrid modeling approach is a combination of microscopic and macroscopic simulation techniques with stochastic modeling. That is, the traffic stream is composed of individual vehicles, while the interactions in the traffic stream are modeled macroscopically using the average speed-density relation. For the future research, however, we need to model some details of signalized operation at junctions by suitably constraining the interlink transfers.

### 6.1 Areas of Future Research

The availability of parallel processing suggests that we develop parallel simulation models for large-scale traffic networks. The event-driven simulation offers an effective working environment to the development of parallel traffic simulation models.

We can usually partition the network into a set of physical processes. A physical process represents a source, a destination, or a group of nodes which consist of one intersection and some dummy nodes. Each physical process $PP_i$ is simulated by a computer program called **logical process** $LP_i$, which maintains the **pool of pending local events** $\Pi_i$. All interactions between physical processes are modeled by the exchange of messages between the corresponding logical processes. Each $LP_i$ is allocated by one processing element $PE_i$.

#### 6.1.1 A Concurrent Simulation Algorithm

A **concurrent simulation** is a mapping of a simulation program to a multiprocessor which can communicate through shared memory. The idea is to use a host machine to distribute
each $LP_i$ ($i = 1, 2, ..., N$) to each processor, and take the host as global synchronizer to handle the concurrency problems. We present the following concurrent simulation algorithm based on the \textit{Lubachevsky Bounded Lag} approach [Lubachevsky 1988].

Let $\tau(e)$ denote the time-stamp of event $e$. The \textit{bounded lag restriction} with parameter $B$ is: “If events $e_1$ and $e_2$ are processed concurrently then $|\tau(e_1) - \tau(e_2)| \leq B$.” The selection of $B$ affects both efficiency and accuracy of the algorithm, where $B$ is any input non-negative constant. Usually $B \leq \min_i T_{i,1}^F$, where $T_{i,1}^F$ is the free travel time on $L_{i,1}$, the first segment of link $L_i$. To maintain the bounded lag restriction, the simulation floor is set to $\min_{e \in \Pi_i} \tau(e)$. An event $e$ is admitted for processing only if $\tau(e) \leq \text{floor} + B$.

Let register $t_i$ represent the local simulated time of each $\Pi_i$. Thus, for each $i$, any $e \in \Pi_i$ satisfies: $\tau(e) \geq t_i$. The concurrent simulation algorithm is described below:

\begin{enumerate}
\item Each processor sends a message to the host with the time-stamp of the next event on its pending event pool;
\item The host selects the minimum time among these event messages as (new) simulation floor, broadcasts this floor to all processors, and synchronizes;
\item Each processor executes the events that satisfy the bounded lag restriction within its pending local event pool, updates its local states, and sends the appropriate messages to its successors.
\item After a processor finishes executing all the actions at this floor, it sends a completion message to all its successors.
\item Perform the \textit{synchronization} statement and let each processor go back to step 1 after it has received all completion messages from all its predecessors.
\end{enumerate}

The simulation starts with $\text{floor} = 0.0$, one vehicle in each source, and $t_i = 0.0$ for all $i$. The \textit{synchronization} statement means that when a processor reaches it then the processor must wait until the other related processors have reached the same statement. A \textit{successor} receives messages from an originating processor, also called its \textit{predecessors}. The loop will not terminate until the local simulation time of each processor equals or exceeds the \textit{termination time}. 

56
Implementations and evaluation of this algorithm is required. However, the potential for a speed-up while using this strategy is limited because each local-event-pool processing bottlenecks a large-scale network. The challenge is to find massively parallel ways to operate on all states in the traffic network to accelerate the simulation [Greenberg et al. 1991].

6.1.2 An Optimistic Distributed Simulation Approach

A distributed simulation is a mapping of a simulation program to a set of processors which can communicate by passing message. One distributed algorithm is based on the optimistic distributed approach that has achieved significant successes across a wide range of application by using the Time Warp mechanism [Jefferson 1985]. An LP can process every message as soon as it arrives; however, if a message with an earlier time-stamp subsequently arrives, the LP must roll back its state to the time of the earlier message and re-execute from that point. This may require the cancellation of message that the LP has sent, which can cause rollback at other LPs. The separation of events and states lends itself well to store the state information which is required for rollbacks and re-execution of some events. Since there exists a set of dummy nodes in some PPs, whose LPs must maintain enough events in their local pending pools as to reduce causality errors. Therefore, we may expect that there are few rollbacks in this distributed approach and thus great speed-up will be achieved.

6.1.3 A Visual Traffic Simulation Support Environment

Another important area is the development of a visual traffic simulation support environment (VTSSE). Visual Simulation is the process of building a model that permits a visual display of its input, internal behavior, and/or output, and experimenting with this model on a computer for a specific purpose [Bishop and Balci 1990]. The purpose of the development of VTSSE is to assist transport professionals to: (1) graphically design a traffic simulation model and its visualization, (2) interactively specify the model's logic, (3) automatically generate the executable version of the model in C programming language, and (4) support
simulation-concurrent animation. Derrick [1992] has developed a prototype of the Visual Simulation Support Environment (VSSE) on a Sun workstation, which contains these five modules: Model Generator, Model Analyzer, Model Verifier, Model Translator, and Visual Simulator. VTSSE can be developed based on the logic of VSSE by restricting VSSE into network-oriented traffic applications and adding an intelligent module called (Knowledge-based) Model Expert. Model Expert is a very useful tool to assist the user to quickly build a valid model. For example, Model Expert should support the optimal selection of an event-set algorithm such that a simulation model in VTSSE is independent of the size of highway network. In an event-driven simulation, an event-set algorithm is required to ensure that the events occur in the proper order and at the proper time. There are two operations that are performed frequently on this set: one to insert new events in the set and one to find the next event and remove it from the set. The choice of the data structure used to maintain this set affects the processor time required for the two operations. Generally [Reeves 1984], (1) the simple linked list is the most efficient alternative if the number of events was small (less than 20 events); (2) the index linear list [Vaucher et al. 1975] is the best for event sets of sizes 20 to 120; and (3) the heap [Wyman 1975] and calendar queue [Brown 1988] are the most efficient for larger sets.

6.1.4 A Real-Time Traffic Simulation

ATMS needs to provide the real-time route guidance information to on-road drivers. This poses a challenge to developing a real-time traffic simulation. For a real-time simulation program, it is not sufficient for the program to be logically correct, the program must also satisfy specific timing requirements. Timing requirements include things such as the maximum allowable time delay between a specified input and the responding output action. However, it is very difficult to design and implement traffic simulations which will guarantee that the appropriate output will be generated at the appropriate times under all possible traffic conditions. This leads to the simulation termination problem [Abrams et al. 1991]:
Given a nonterminating simulation problem and a termination condition, specify how program may be terminated at the simulation time that satisfies the termination condition. It consists of three steps:

1. Find a value of simulation time $t$ when the termination condition is satisfied,
2. Collect and report output measures at that time $t$, and
3. Stop execution of the simulation.

Anyway, real-time traffic models are non-trivial and expensive systems to develop. They usually require special hardware for monitoring and dynamic display. In these systems the computer is usually interfaced directly to the physical equipment in the real network and is dedicated to monitoring or controlling the operation of that equipment. An operator’s console is required for manual intervention. The human operator is kept constantly informed of the state of the transportation system by displays of various types including graphical ones. Records of the system’s state changes are also kept in an information base which can be interrogated by the operators at will, to support decision making in the day-to-day running of systems [Buras et al. 1990].
REFERENCES


VITA

Tungsheng Yu entered this world on a cold November morning in 1964 and found himself in the sleepy village near Ningbo, China. He led a simple life there until 1980 when his father’s work took the family to his dreamed city of Ningbo. He attended Peking University where he graduated with a MS in Applied Mathematics in July 1989. After graduation he was employed by Ningbo Computer Center, Ningbo, China. In July 1990 Mr. Yu arrived at USA and joined a PhD program in Mathematics at Virginia Polytechnic Institute and State University. In May 1991 he transferred in Computer Science at the same university. He received MS in Computer Science in September 1992.